



Université du Québec  
**École de technologie supérieure**

# LOG792 : PROJET DE FIN D'ÉTUDE EN GÉNIE LOGICIEL

Présenté à

ALAIN APRIL ET CHRISTIAN DESROSIERS

Département de génie logiciel et des technologies d'information

---

## EndoMine

---

David Lauzon

LAUD01028300

Anton Zakharov

ZAKA12038406

20 décembre 2012

# Remerciements

Ce projet a pris naissance grâce à l'engagement du professeur Dr. Alain April dans le milieu de la santé .De plus, Dr. Alain April et Dr. Christian Desrosier nous ont fourni beaucoup d'aide tant du point de vue scientifique et que du point de vue des relations avec les clients. Sans leurs aides le projet EndoMine aurait été impossible à réaliser.

Un gros merci au Dr. Shaun Eintracht et Dr. Elizabeth MacNamara, qui ont activement participé au projet et qui nous ont fourni tous les détails nécessaires afin de réaliser un projet répondants à leurs besoins. Nous espérons que le projet leur serait utile dans le futur.

Finalement, nous remercions Dr. Marc Trifiro, Dr. Michael Miriello, Stéphane Benhamou, Josie Trevisonno, Roberta Maizen, Angelo, Samplis, Reisha Melnick, Chris Polly Kandriotis de nous avoir aidé tout au long du projet.

# Résumé

Le but de ce projet est d'évaluer les besoins en forage de données des clients des Centres Endocrinologie et de Diagnostic Médical. Par la suite, notre objectif est de proposer une solution de forage de données optimisée par rapport à la solution existante. Dans le cadre de ce projet, nous nous sommes concentrés sur les besoins du Dr. Shaun Eintracht. Une étude technologie avait été effectuée et deux solutions ont été présentées : une utilisant le modèle relationnel avec la base de données Oracle, et l'autre en utilisant la technologie "BigData" Impala. L'architecture, les prototypes et les comparaisons de 2 solutions ont été fournis, ainsi qu'une liste de recommandations.

# Table des matières

<b>Répartition des tâches pour le rapport final</b>	<b>11</b>
<b>Introduction</b>	<b>12</b>
<b>1 Phase I : Inception</b>	<b>13</b>
1.1 Besoins d'affaires . . . . .	13
1.2 Besoins d'affaire . . . . .	14
1.3 Gestion des risques . . . . .	14
1.4 Exploration technologique . . . . .	19
1.4.1 Bases de données relationnelles . . . . .	20
1.4.1.1 oracle . . . . .	20
1.4.2 Bases de données NoSQL Open Source . . . . .	20
1.4.2.1 Hive . . . . .	21
1.4.2.2 HBase . . . . .	21
1.4.2.3 Impala . . . . .	22
1.4.3 Pentaho . . . . .	23
1.4.4 Technologies Propriétaires . . . . .	23
1.5 Analyse et conception du système . . . . .	24
1.5.1 Infrastructure du système . . . . .	24
1.5.2 Tables Utilisées . . . . .	24

<i>TABLE DES MATIÈRES</i>	4
1.5.3 Tables Anonymisées . . . . .	27
1.6 Anonymisation des données . . . . .	29
1.7 Statistiques sur les Tables . . . . .	29
<b>2 Choix Techonologiques</b>	<b>30</b>
<b>3 Phase II-a : Prototype Hadoop</b>	<b>31</b>
3.1 Installation . . . . .	31
3.1.1 Méthode d'installation . . . . .	31
3.1.1.1 Choix du système d'exploitation . . . . .	31
3.1.1.2 Installation avec le Cloudera Manager . . . . .	32
3.1.1.2.1 Avantages du Cloudera Manager . . . . .	32
3.1.1.2.2 Désavantages du Cloudera Manager . . . . .	32
3.1.1.3 Installation avec des rpm . . . . .	33
3.1.1.3.1 Scripts init.d pour Impala . . . . .	33
3.1.2 Configuration . . . . .	33
3.1.2.1 Configuration du matériel . . . . .	33
3.1.3 Configuration du kernel Linux . . . . .	34
3.1.4 Configuration de Hadoop . . . . .	34
3.2 Architecture logicielle . . . . .	35
3.2.1 Architecture de Hadoop . . . . .	35
3.2.1.1 Architecture de HDFS . . . . .	35
3.2.1.2 Architecture de Map-Reduce . . . . .	36
3.2.2 Architecture de Hive . . . . .	36
3.2.3 Architecture d'Impala . . . . .	36
3.2.4 Importation des données . . . . .	37
3.2.4.1 Importation avec Quest Oracle Data Connector . . . . .	38

<i>TABLE DES MATIÈRES</i>	5
3.2.5 Compression . . . . .	38
3.2.6 Partionnement . . . . .	39
3.2.7 Dé-normalisation . . . . .	40
3.2.8 Coordination des opérations . . . . .	42
3.3 Problèmes rencontrés . . . . .	45
3.4 Performance . . . . .	46
3.5 Interface visuelle . . . . .	46
<b>4 Phase II-b : Prototype Oracle</b>	<b>47</b>
4.1 Installation . . . . .	47
4.2 Architecture logicielle . . . . .	48
4.2.1 Denormalisation . . . . .	48
4.2.2 Clustering . . . . .	49
4.2.3 Partitionnement . . . . .	49
4.2.4 Compression . . . . .	51
4.3 Problèmes rencontrés . . . . .	51
4.4 Performance . . . . .	52
4.5 Interface visuelle . . . . .	53
<b>5 Comparaison des deux solutions</b>	<b>54</b>
<b>Conclusion</b>	<b>58</b>
<b>Recommandations</b>	<b>59</b>
<b>Liste de références</b>	<b>61</b>
<b>Bibliographie</b>	<b>83</b>
<b>A Plan de travail</b>	<b>85</b>

<i>TABLE DES MATIÈRES</i>	6
<b>B Document de Vision</b>	<b>97</b>
<b>C Document de SRS</b>	<b>119</b>
<b>D Document d'architecture (brouillon)</b>	<b>145</b>
<b>E Compte-rendus de réunions</b>	<b>153</b>
<b>F Rapport de résolution des incidents</b>	<b>174</b>
<b>G Connaissances en lien avec le bac de génie logiciel</b>	<b>187</b>
<b>H Survol des produits Pentaho</b>	<b>194</b>
<b>I Installation de Hadoop</b>	<b>195</b>
<b>J Configuration de Hadoop</b>	<b>210</b>
<b>K Code SQL pour Oracle</b>	<b>223</b>
<b>L License BSD</b>	<b>226</b>

# Liste des tableaux

1	Abréviations, sigles et acronymes . . . . .	9
2	Répartition des tâches pour le rapport final . . . . .	11
1.1	Analyse des Risques . . . . .	15
1.2	Statistiques sur les Tables . . . . .	29
A.1	Échéancier . . . . .	85
G.1	Grille de relations entre ce PFE et les connaissances de génie logiciel . . . . .	187

# Table des figures

1.1	Clé dans HBase . . . . .	22
1.2	Perspective du produit . . . . .	25
1.3	Modèle relationnel des tables du projet . . . . .	26
1.4	Modèle relationnel anonymisé des tables du projet . . . . .	28
3.1	Architecture d'Impala . . . . .	36
3.2	Importation des tables dans Hadoop . . . . .	37
3.3	Partionnement des tables dans Hive . . . . .	40
3.4	Dénormalisation des tables dans Hive . . . . .	41
3.5	Interface principale de Jenkins . . . . .	43
3.6	Vue d'une job de Jenkins . . . . .	43
3.7	Configuration d'une job de Jenkins . . . . .	44
3.8	Preuve que le système "swappait" . . . . .	46
5.1	Des requêtes de jointure et de copie. . . . .	55
5.2	Impact de compression avec différent types de fichiers. . . . .	55
5.3	Requêtes sur les données réelles. . . . .	56
H.1	Mindmap des produits Pentaho . . . . .	194

# LISTE DES ABRÉVIATIONS, SIGLES ET ACRONYMES

Cette section contient une liste d'abréviations dont la définition semble du point de vue auteur la plus utile au lecteur.

TABLE 1: Abréviations, sigles et acronymes

Abréviation	Description
OLAP	Online analytical processing
Hadoop	Système de fichier distribué
RDBMS	Base de données relationnelles (Relational Database Management System)
JGH	Jewish General Hospital
Environnement staging	Environnement servant à traiter les données avant de les envoyer au système analytique
TABLESPACE	Espace physique où se retrouvent les tables
full table scans	Activité de balayer une table complètement du début à la fin

TABLE 1: (suite)

Cluster	Façon de regrouper les données dans Oracle sans redondance de plusieurs tables dans une même table en utilisant une clé commune
Mb	Mega Octets
Gb	Giga Octets
Kb	Kilo Octets
PAGE FAULTS	Quand le système accède les données dans la mémoire virtuelle, au lieu de la mémoire vive.
Ad hoc	Une requête improvisée
SQL	Structured Query Language

# Répartition des tâches

TABLE 2: Répartition des tâches pour le rapport final

Chapitre	Section	Auteur
	Modèle LaTeX	David Lauzon
	Introduction	Anton Zakharov
Démarrage du projet	Toutes les sections	David Lauzon
Phase I : Inception	Toutes les sections	Anton Zakharov
Gestion des risques	Toutes les sections	Anton Zakharov
Division de la Phase II	Toutes les sections	Anton Zakharov
Phase II-a	Prototype Hadoop	Anton Zakharov
Phase II-b	Prototype Oracle	David Lauzon
Comparaison des deux solutions	Toutes les sections	David Lauzon
Conclusion	Toutes les sections	Anton Zakharov
Recommandations	Toutes les sections	David Lauzon
Liste de références	Toutes les sections	David Lauzon
Plan de travail	Toutes les sections	David Lauzon

# Introduction

Un des sujets les plus chauds de l'actualité dans le domaine de forage de données est le "BigData". L'idée derrière le "BigData" est de pouvoir traiter un grand volume de données, sur du "hardware" qui ne coûte pas cher. Beaucoup d'entreprises désirant faire du forage de données sont très intéressées par la technologie. Par contre, ils ne savent pas nécessairement quels sont les avantages et inconvénients des solutions "BigData". Sans conter qu'il peut être très difficile sans avis d'expert de savoir si une solution "BigData" particulière peut aider à résoudre le problème du client.

Dans le cadre de ce projet, la problématique du client est une base de données non optimisée pour faire des requêtes analytiques. Voyant large, le client voudrait qu'on lui propose une solution permettant de faire des requêtes analytiques facilement et rapidement sur un système qui peut s'adapter à faible coût à un volume de données grandissant.

Ce projet a comme objectif d'analyser et comparer les deux solutions les plus adaptées aux besoins du client. La première consisterait en une base de données Oracle optimisée, et l'autre en un système "BigData" Impala. Des tableaux de comparaison ainsi que des prototypes seraient fournis au client afin qu'il puisse faire des choix éclairés.

# Chapitre 1

## Phase I : Inception

### 1.1 Besoins d'affaires

Les besoins d'affaires ainsi que les requis sont décrits précisément dans les documents Vision et SRS. En résumé, il y a trois besoins principaux catégorisés par intervenant :

- Les besoins du Dr. Shaun Eintracht incluent le pouvoir de faire des Requêtes de type "ad hoc". Les problèmes principaux avec le système existant étaient que le système était lent et que les recherches pouvaient seulement être faites sur un sous ensemble de données. De plus, les recherches ne devraient pas être effectuées sur la base de données opérationnelle en production comme c'est le cas actuellement. Dr. Shawn n'étant pas un expert en SQL, un générateur de requêtes de type Microsoft Access serait essentiel.
- Le Dr. Elizabeth MacNamura a besoin d'un système lui permettant une meilleure gestion des coûts et ressources.
- Le Dr. Mark Trifiro , ainsi que les autres intervenants voudraient ulté-

rieurement un système permettant le forage de données sur toutes les données médicales de tous les départements du JGH.

Considérant le temps limité dont nous disposions, nous avons décidé de nous concentrer sur les besoins du Dr. Shawn Eintracht.

## **1.2 Besoins d'affaire**

Les besoins d'affaires ainsi que les réquis sont décrits précisément dans les documents Vision et SRS. Un cour résumé

## **1.3 Gestion des risques**

Cette section décrit l'évolution des risques associés au projet. Voir la Table 1.1.

TABLE 1.1: Analyse des Risques

Risque	Impact	Proba- bilité	Mitigation / Atténuation)
R1. La portée (« scope ») du projet est très ambitieuse	<b>Faible</b>	<b>Faible</b>	<ul style="list-style-type: none"> <li>– L'emphase du projet sera sur la documentation précise, ce qui facilitera les prochaines équipes qui travailleront sur le projet.</li> <li>– Le projet sera réalisé en plusieurs courtes itérations afin de pouvoir obtenir régulièrement de la rétroaction de la part du client</li> <li>– Le prototype réalisé de répondra qu'à un seul besoin du client, ce qui fournira une preuve de concept et offrira un minimum de fonctionnalités au client.</li> <li>– <b>Nous avons diminué la portée afin de répondre au besoin du Dr. Eintracht seulement.</b></li> </ul>

TABLE 1.1: (suite)

Risque	Impact	Proba- bilité	Mitigation / Atténuation)
R2. La confidentialité des données est compromise	Élevé	Faible	Afin de réduire le risque que les données confidentielles sur les patients soient divulguées, les informations personnelles sur les patients seront omises du système «Hadoop». Seul le Dr. Shaun aura accès aux données confidentielles.
R3. Les technologies utilisées sont mal connues	<b>Faible</b>	<b>Faible</b>	<ul style="list-style-type: none"> <li>– Lire la documentation et faire des exercices. De plus, nous considérons la possibilité d’assister à la conférence «Hadoop World» en octobre, ce qui inclut une activité de formation.</li> <li>– <b>Nous avons une bonne idée des limitations des technologies.</b></li> </ul>

TABLE 1.1: (suite)

Risque	Impact	Proba- bilité	Mitigation / Atténuation)
R4. L'écosystème «Hadoop» ne répond pas aux besoins du client	Moyen	Moyen	<ul style="list-style-type: none"> <li>- Nous allons commencer à s'informer sur «Hadoop» le plus tôt possible. De plus, nous allons collaborer étroitement avec le client pour nous assurer qu'il comprend la solution et qu'il reçoit la formation nécessaire.</li> <li>- <b>Les solutions Impala/Hive semblent répondre aux besoins du client, mais une certaine adaptation de la part du client est nécessaire</b></li> </ul>
R5. L'importation des données est inexacte	Élevé	Faible	<ul style="list-style-type: none"> <li>- Nous allons développer un bon jeu de test, pour s'assurer que les données sont bien importées.</li> <li>- <b>Pour le prototype cette partie serait réalisée partiellement.</b></li> </ul>

TABLE 1.1: (suite)

Risque	Impact	Proba- bilité	Mitigation / Atténuation)
R6. Le système développé est moins performant que la solution Oracle existante	<b>Moyen</b>	<b>Élevé</b>	<ul style="list-style-type: none"> <li>– Des discussions importantes avec le client seront réalisées afin de s’assurer que le système développé soit plus performant que le système existant.</li> <li>– <b>Avec une seule machine, c’est certain. Les systèmes distribués sont faits pour marcher avec plusieurs machines. Donc à court terme c’est inévitable.</b></li> </ul>

TABLE 1.1: (suite)

Risque	Impact	Proba- bilité	Mitigation / Atténuation)
R7. Les besoins du client sont mal compris	<b>Faible</b>	Faible	<ul style="list-style-type: none"> <li>– Nous avons planifié des rencontres hebdomadaires avec le client, en plus de réaliser plusieurs documents précisant le projet de manière itérative (Vision, SRS, Cas d'utilisation, etc.). Chaque document sera présenté au client afin de collecter une rétroaction permettant d'ajuster le projet aux besoins du client.</li> <li>– <b>Avec toutes les rencontres effectués, ainsi que les documents de vision et SRS signés, il est très peu probable que les besoins soient mal compris. Par contre, seule une partie des besoins est réalisable.</b></li> </ul>

## 1.4 Exploration technologique

Suite à l'étude des besoins et des risques, nous avons effectué une étude technologique. Voici les technologies explorées, leurs descriptions et les choix

que nous avons effectués.

## 1.4.1 Bases de données relationnelles

### 1.4.1.1 oracle

Avant de réinventer la roue, nous avons pensé à utiliser les bases de données existantes. Comme la base de données Oracle, existante en production n'est pas optimisée pour faire des requêtes analytiques, il nous encombre de le faire.

Il y a plusieurs raisons de créer un prototype de la BD Oracle. Premièrement, nous n'avons pas vraiment le choix, puisque la base de données de "staging" devrait être la même que celle de production. Deuxièmement à cause des risques R4 et R6, il faudrait au moins faire quelques tests pour confirmer la validité d'une solution Open Source en termes de performance. Finalement étant donné que l'hôpital a déjà plusieurs bases de données oracle, le support et la maintenance de cette solution présenteraient un avantage additionnel. Par contre, il n'y a aucune garantie en terme de mise à l'échelle et cette solution devrait donc être temporelle.

## 1.4.2 Bases de données NoSQL Open Source

Les bases de données NoSQL dont on va présenter vont être bâties en utilisant le système de gestion de fichier distribué «Hadoop». Présenter l'architecture de «Hadoop» va au-delà de la portée du présent document, mais si vous êtes intéressé voici un lien qui la décrit assez sommairement : [http://en.wikipedia.org/wiki/Apache\\_Hadoop](http://en.wikipedia.org/wiki/Apache_Hadoop).

De base les données sont stockées dans des fichiers qui sont automatiquement répliqués sur plusieurs disques. MapReduce est un "framework" utilisé

pour créer des tâches fonctionnent en parallèle et permettant de faire des opérations sur les fichiers stockés dans «Hadoop». Durant l'étape « Map », le nœud principal (master) prend les données d'entrées, les divise en problèmes plus petits, et les distribue aux nœuds de travail. Durant l'étape «Reduce» le nœud principal collecte les résultats de tous les sous-problèmes et les combine dans un résultat. Ce résultat est une réponse au problème posé initialement. Voir <http://en.wikipedia.org/wiki/MapReduce> pour plus de détails.

#### 1.4.2.1 Hive

Hive est une infrastructure de «data warehouse» construite en top de « Hadoop ». Hive permet de faire de l'analyse, des résumés et des requêtes dans «Hadoop» ([http://en.wikipedia.org/wiki/Apache\\_Hive](http://en.wikipedia.org/wiki/Apache_Hive)) avec un langage ressemblant de beaucoup à SQL. Ceci fait en sorte que c'est une solution très simple à implémenter.

Le problème avec Hive est qu'il utilise les fonctions MapReduce qui sont très lentes à commencer (aux alentours de 20 secs!!! selon nos tests). De plus, les jointures qui peuvent être faites sont limitées. Si jamais la structure des données de la BD change, les données devraient être réimportées complètement d'Oracle. Hive ne supporte pas (facilement) les opérations de mise à jour ou d'effacement des données.

#### 1.4.2.2 HBase

HBase est une infrastructure par-dessus Hadoop, mais qui n'utilise pas MapReduce. HBase store les tables par colonnes (alors que traditionnellement ils sont stockés par rangées), ce qui lui permet de réduire grandement le temps de recherche (car on ne lit que les informations des colonnes dont on à

besoin). HBase supporte les "updates" et les suppressions naturellement (avec son système de Versionnement). Par contre, il n'y a aucun moyen simple et efficace de faire des jointures avec HBase.

La figure 1.1 décrit la structure d'une clé HBase.

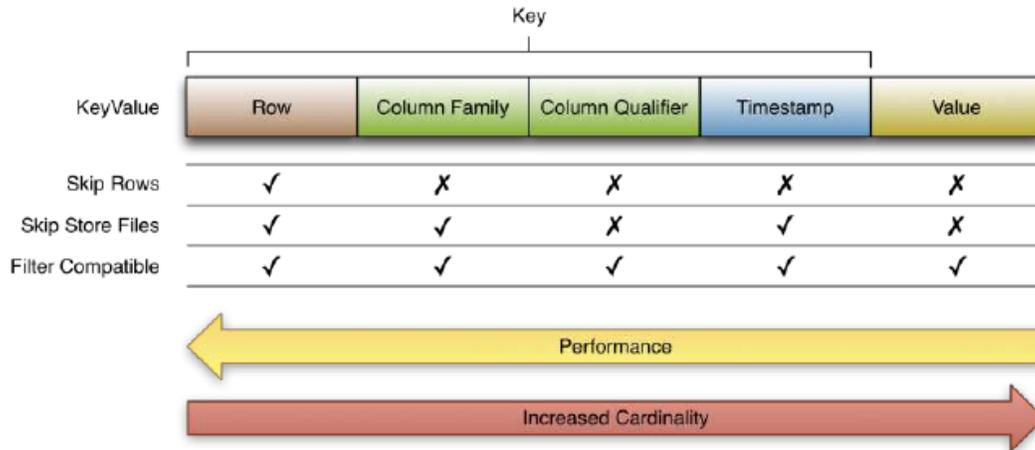


FIGURE 1.1 – Clé dans HBase  
Source : (George, 2011)

Faire des recherches en connaissant la valeur d'une clé, ou d'un préfixe de clé est très rapide. Tous les autres types de recherche vont demander de faire un "full table scan", qui est plus inefficace dans HBase que dans Hive.

Il serait très difficile d'utiliser cette solution directement pour le Dr.Eintracht, car ce dernier a besoin de rechercher en utilisant comme critère plusieurs colonnes sans ordre prédéfini.

### 1.4.2.3 Impala

Le problème principal de Hive est le « start up cost » à cause de l'utilisation du MapReduce. Impala est un projet séparé développé par Cloudera. L'objectif est d'être un engin de recherche en temps réel pour «Hadoop» (<http://www.>

`theregister.co.uk/2012/10/24/cloudera_hadoop_impala_real_time_query/`)

. En gros on peut le voir comme Hive, mais plus rapide (important à noter il n'a pas toutes les possibilités de Hive mais dans notre cas d'utilisation on peut considérer comme tel. Voici le lien du papier de google sur lequel il est basé : [http://static.googleusercontent.com/external\\_content/untrusted\\_dlcp/research.google.com/en//pubs/archive/38125.pdf](http://static.googleusercontent.com/external_content/untrusted_dlcp/research.google.com/en//pubs/archive/38125.pdf)).

Le projet est encore en stage bêta. Par contre, il offre des promesses intéressantes, et il est possible qu'on le teste.

### 1.4.3 Pentaho

Pentaho est un outil open source, pouvant intégrer diverses sources de données ainsi que de faire des solutions de BI (Business Intelligence). Nous avons essayé les fonctions d'intégration et ils marchent plutôt bien. Le GUI pour les solutions BI n'est pas gratuit dans les majeures parties des cas. Après analyse nous avons déterminé que les solutions BI ne permettant pas de faire toutes les requêtes "ad hoc", ce qui fait que cette solution avait été rejetée.

Nous aimerions ajouter qu'avant de rejeter cette solution, nous avons fait un survol de tous les produits offerts par Pentaho et construit un "mindmap" (voir l'annexe H).

### 1.4.4 Technologies Propriétaires

Nous avons étudié les technologies suivantes : **Datameer**, **Vertica**, **Greenplum**, **Teradata**. Par contre, vu les contraintes budgétaires et la portée du projet, nous n'avons retenu aucune de ces solutions.

## 1.5 Analyse et conception du système

Cette section décrit du point de vue conceptuel l'architecture générique pour les deux solutions. L'architecture spécifique à chaque prototype serait définie dans la Phase II.

### 1.5.1 Infrastructure du système

La figure décrit la perspective haut niveau du produit. Pour plus de détails, voir le document de vision dans l'Annexe. En résumé l'environnement "staging" contiendrait les tables utiles extraites de la BD de production. Le pré-traitement et l'anonymisation des tables seraient faits dans cet environnement. L'environnement EndoMine contiendrait la solution proposée au client. L'environnement "Restricted" ne serait pas développé dans le cadre de ce projet.

### 1.5.2 Tables Utilisées

Le nombre de tables dans la base de données est énorme. Comme nous ne faisons qu'un prototype, importer toutes les tables avec toutes les relations dépasse la portée de nos possibilités. Nous avons donc extrait et traité les tables principales que le Dr. Eintracht allait utiliser.

La figure 1.3 représente les tables et leurs relations.

Les tables commençantes par **v\_p** contiennent les données particulières, alors que les **v\_s** servent à décrire l'information dans les tables **v\_p**. Par exemple, **v\_s\_lab\_test** contient le nom du test et les caractéristiques propres servant à classifier un test par rapport aux autres. **v\_p\_lab\_test\_result** contient des données comme les dates des tests réalisés, ainsi que les valeurs

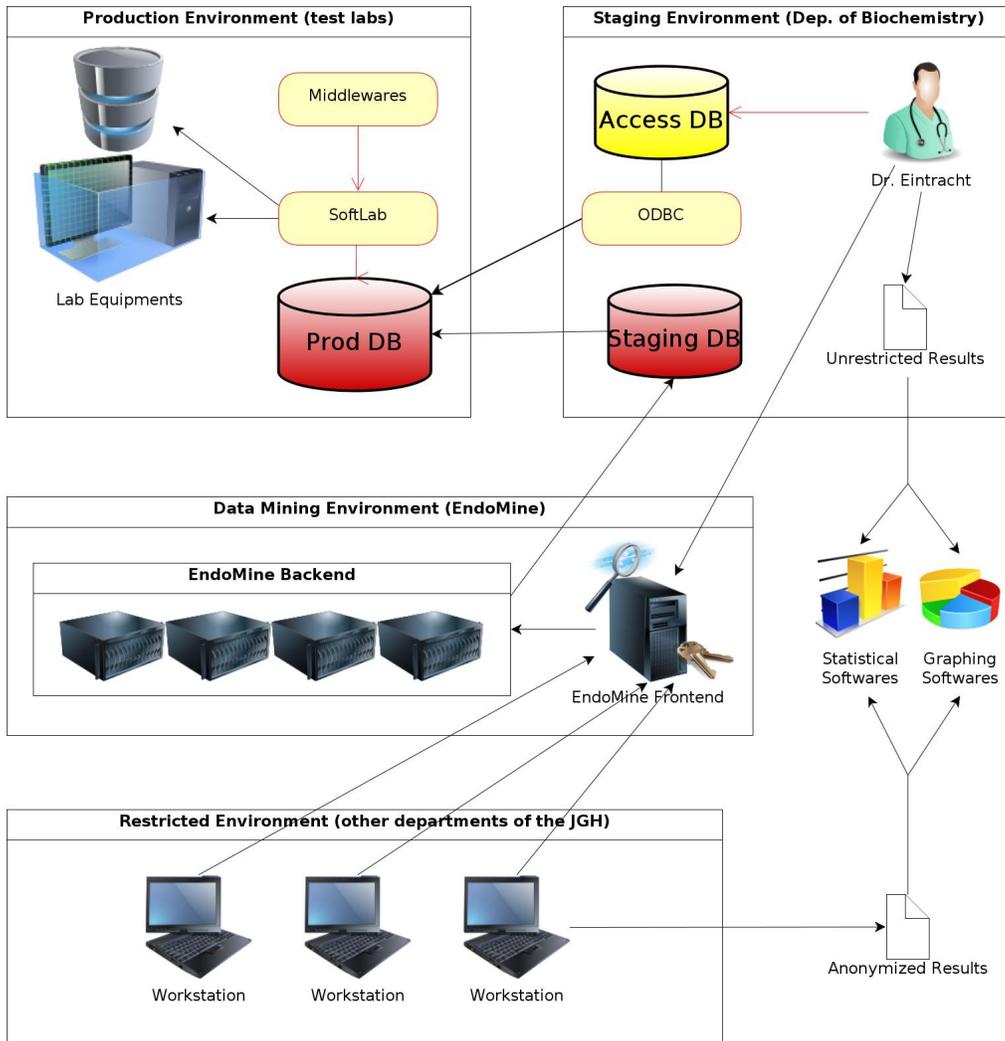


FIGURE 1.2 – Perspective du produit

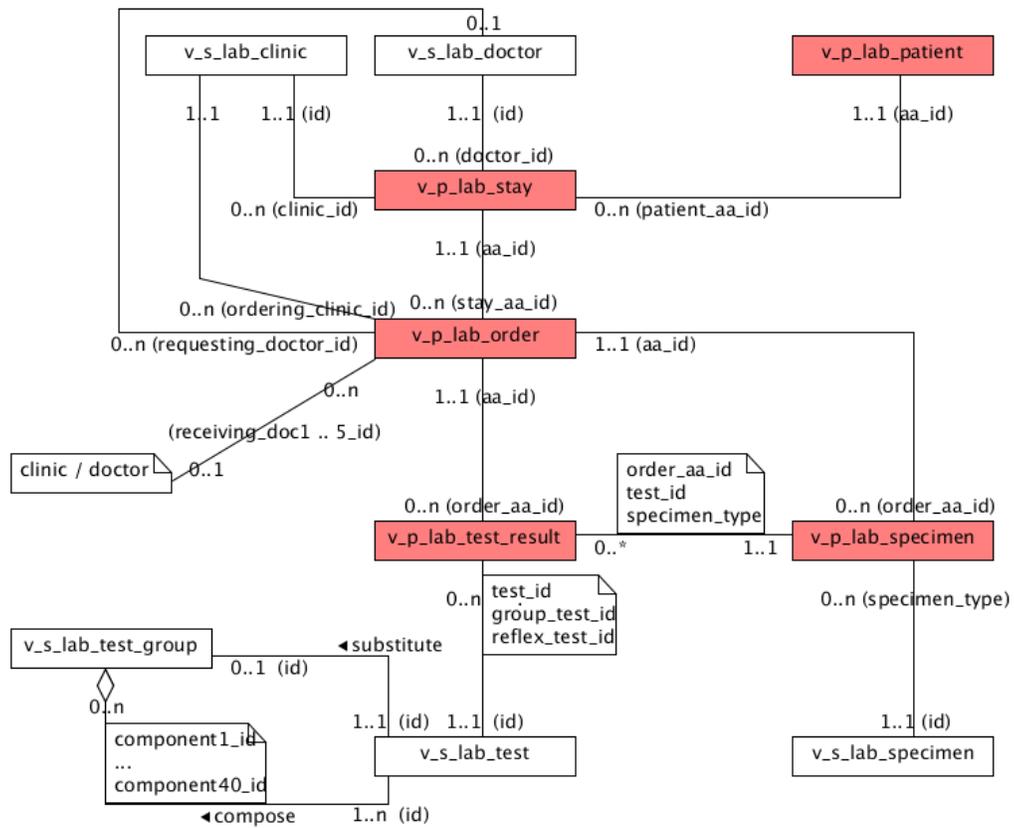


FIGURE 1.3 – Modèle relationnel des tables du projet

reliées aux tests.

Voici la description des tables :

- **v\_p\_lab\_patient** contient les données des patients qui visitent l'hôpital. Cette table serait anonymisée.
- **v\_p\_lab\_stay** contient les séjours de patients dans l'hôpital.
- **v\_p\_lab\_order** contient la commande de tests effectués durant un séjour particulier.
- **v\_p\_lab\_test\_result** contient les résultats des tests.
- **v\_p\_lab\_specimen** contient les informations spécimens physiques (ex : urine) utilisés pour faire des tests.
- **v\_s\_lab\_test\_group** décrit les composantes d'un test particulier.
- **v\_s\_lab\_test** contient des informations supplémentaires sur chaque type de test.
- **v\_s\_lab\_specimen** contient des informations communes sur chaque type de spécimen.
- **v\_s\_lab\_doctor** contient les informations sur les docteurs.
- **v\_s\_lab\_clinic** contient les informations sur les cliniques.

D'autres tables comme **v\_p\_lab\_message** devraient être intégrées dans la prochaine version du logiciel.

### 1.5.3 Tables Anonymisées

Étant donné que la table **v\_p\_lab\_patient** contient des données confidentielles, elle serait anonymisée. De plus, afin d'augmenter la vitesse du traitement, les tables **v\_p\_lab\_patient\_anon** (patient anonymisé) et **v\_p\_lab\_stay** seront fusionnées dans la même table : **v\_p\_lab\_pat\_stay\_anon**. La figure 1.4 représente les tables anonymisées et leurs relations.

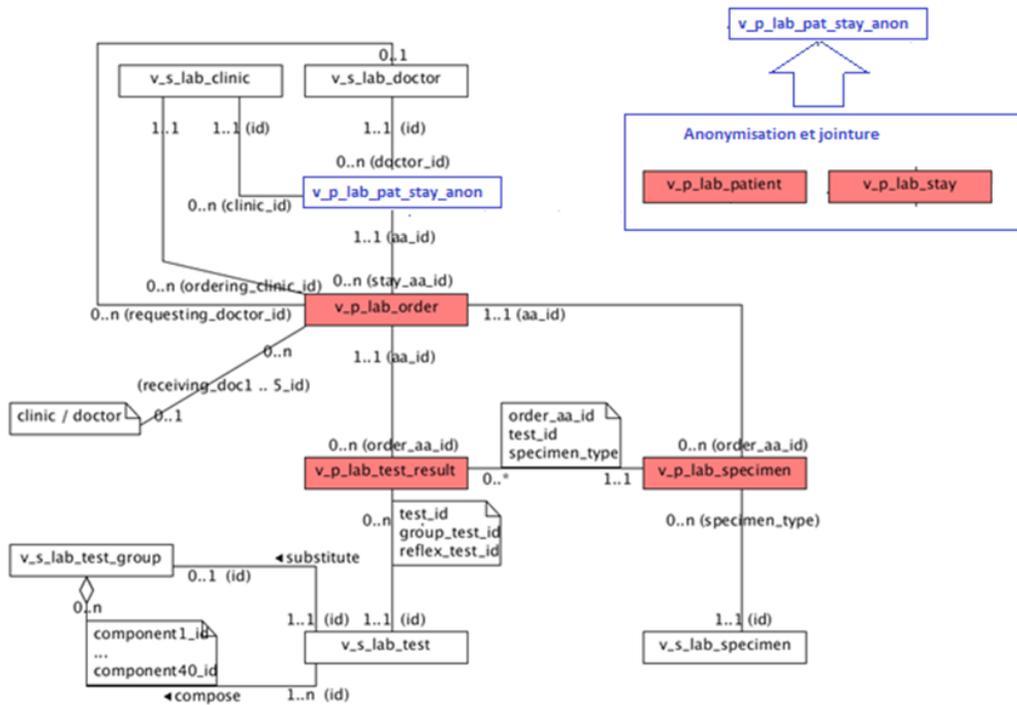


FIGURE 1.4 – Modèle relationnel anonymisé des tables du projet

La table `v_p_lab_pat_stay_anon` contient les informations de la table `v_p_lab_stay`, une clé anonymisée, le sexe et la date de la naissance du patient. La table `v_p_lab_patient_link` pourrait être accédé par un superutilisateur (ex : Dr. Shawn Eintracht) et servirait de faire le lien entre les patients anonymisés et les patients réels.

## 1.6 Anonymisation des données

Le processus d'anonymisation est décrit dans le document d'architecture présent dans l'annexe. Les scripts `create_anon_tables.sql` et `fill_anon_tables.sql` créent et peuplent les tables anonymisées. La création prend environ 5s (sur le laptop de A.Z.), alors que le remplissage prend 155 secondes (sur le laptop de A.Z.).

## 1.7 Statistiques sur les Tables

La table 1.2 décrit la taille des principales tables. Étant donné que la table `v_p_lab_message` n'était pas incluse dans cette version du projet, nous n'allons pas fournir des statistiques.

TABLE 1.2: Statistiques sur les Tables

Tables	Nb. Rangées (Millions)	Taille (GB)
<code>v_p_lab_stay</code>	2.6	0.3
<code>v_p_lab_order</code>	10.4	2.1
<code>v_p_lab_specimen</code>	23.9	2.3
<code>v_p_lab_test_results</code>	318.1	55.5

# Chapitre 2

## Choix Technologiques

En considérant les recommandations de la section 1.4, nous avons décidé de créer un prototype basé sur la technologie Impala de Hadoop.

Nous avons décidé de créer également une solution Oracle, afin de comparer les performances. De plus considérant, la facilité de déploiement et la maintenance pour le département d'opération actuel de l'hôpital, cette solution pourrait, si les performances sont acceptables, être une solution temporaire. Il est à noter que d'autres RDBMS gratuits auraient pu faire l'affaire, mais nous avons considéré qu'Oracle était la base de données la plus avancée, et donc la meilleure pour nous fournir les résultats de comparaison.

Alors dans la phase II-a (Chapitre 3), nous allons parler en détail de la solution "Impala", et dans la phase II-b (Chapitre 4), nous allons parler de la solution Oracle. Afin dans le (Chapitre 5) nous allons comparer les deux solutions.

# Chapitre 3

## Phase II-a : Prototype Hadoop

Cette section décrit le prototype basé sur l'éco-système Hadoop. Il existe d'autres solutions NoSQL qui auraient pu être envisagées, mais par contrainte de temps il était impossible de toutes les essayer.

### 3.1 Installation

#### 3.1.1 Méthode d'installation

##### 3.1.1.1 Choix du système d'exploitation

D'abord, nous avons étudié la possibilité d'installer Hadoop sur notre ordinateur personnel, mais très peu de documentation était disponible. Toutes les sources d'informations portaient à croire que les utilisateurs finissent tôt ou tard par migrer à Linux dû à des bogues (Said, 2011). Nous avons donc choisi d'installer Hadoop sur Linux Mint 13 (basé sur Ubuntu), qui était déjà installé sur l'ordinateur de bureau.

Par la suite, nous avons voulu répliquer un environnement Ubuntu portatif (à l'aide d'un laptop et un disque dur externe) afin de faciliter le développement

à la maison et les échanges au JGH. Cependant, les performances étaient atroces et cette idée a vite été abandonnée.

Et lorsque le choix d'utiliser Impala est arrivé, nous n'avons pas réussi à compiler Impala sur Ubuntu/Mint. Comme il s'agissait d'un beta qui venait d'être annoncé, il n'y avait pas d'instructions de compilations et des paquets n'étaient disponibles que pour l'architecture Red Hat. Le choix s'est donc tourné vers CentOS 6.2, la version exacte du système recommandé par les créateurs de Impala.

### **3.1.1.2 Installation avec le Cloudera Manager**

La première version d'Impala que nous avons installé avec succès fut avec le Cloudera Manager, un outil de configuration avec une interface web.

#### **3.1.1.2.1 Avantages du Cloudera Manager**

- Facile de modifier un paramètre de configuration ou installer une librairie supplémentaire sur plusieurs machines en un seul instant.
- Affiche les configurations qui sont reliées à l'option qui est en train d'être modifiée. Peut être utile pour les néophytes à Hadoop.

#### **3.1.1.2.2 Désavantages du Cloudera Manager**

- Quand même nécessaire de faire certaines manipulations à la ligne de commande (configuration de hive, par exemple).
- Certains services doivent absolument être redémarrés à l'aide de l'interface graphique.
- Impossible de changer la configuration SANS l'interface graphique.
- Impossible de changer le nom d'hôte d'un serveur (c.-à-d. de localhost vers le bon nom d'hôte) configuré, contrairement à si on changait ma-

nuellement les fichiers hdfs-site.xml.

### 3.1.1.3 Installation avec des rpm

Pour les 2 dernières raisons mentionnées ci-haut, j'ai ré-installé Impala et CDH4 une quatrième fois. Cette fois j'ai utilisé les rpm, et ça très bien fonctionné. Les détails d'installation sont disponible à l'annexe I.

Il y a 3 fichiers d'installation; un pour chaque composante essentielle : hadoop, hive et impala.

**3.1.1.3.1 Scripts init.d pour Impala** La version 0.2 d'impala ne venait pas avec des scripts init.d. On devait démarrer le impala daemon et le impala state store manuellement. Cependant si on perdait la connexion, on devait constamment relancer ces commandes. Pour cette raison, j'ai créé deux scripts init.d pour chaque daemon d'impala. Voir à la fin de l'annexe I.

## 3.1.2 Configuration

### 3.1.2.1 Configuration du matériel

Le serveur qui a été retenu pour livrer au JGH est un Dell PowerEdge 2950 :

- Dual Xeon 2.33 GHz 1333mhz FSB 4MB L2
- 16 GB RAM
- 4 HDD SCSI 15K (1 x 73 GB @10K, 3 x 300 GB)
- Le serveur n'a pas été virtualisé.
- Le système d'exploitation est installé sur le disque de 73GB.
- Hadoop utilise les 3 autres disques pour les données.
- Pas de système RAID.

### 3.1.3 Configuration du kernel Linux

J'ai fait une petite modification pour améliorer les performances, en changeant le *swappiness* du kernel. Cette configuration indique au kernel d'éviter de *swapper* les processus de la mémoire physique pour aussi longtemps que possible (TheNinjaSysAdmin, 2011). Normalement cette configuration vaut autour de 50, ce qui satisfait la plupart des cas d'utilisation; mais c'est inadéquat pour Hadoop.

```
1 sysctl vm.swappiness=0
```

### 3.1.4 Configuration de Hadoop

Dans Hadoop, beaucoup de problèmes surviennent du au volume de données et la complexité des requêtes effectuées. Tout au long du PFE j'ai du modifier la configuration pour améliorer l'expérience utilisateur.

Tous ces changements ont été annotés à l'aide de commentaires à l'intérieur des fichiers de configurations, que j'ai inclus à l'annexe J. Les fichiers inclus sont : `core-site.xml`, `hdfs-site.xml`, `mapred-site.xml`, `hive-site.xml`, et `hive-env.sh`.

Les configurations les plus importantes sont :

- Augmentation de la taille de block HDFS par défaut de 64 MB à 256 MB. De toute façon, hive doit lire tous les fichiers de la table / partition. Plus les fichiers sont gros, moins il y'a de petits fichiers, et ainsi le NameNode est plus efficace.
- Allocation de la mémoire aux tâches.
- J'ai alloué 3 tâches map et 3 tâches reduce. Normalement on doit en mettre une par coeur donc on pourrait aller jusqu'à 4, mais je préfère en

laisser un coeur de disponible pour les autres services (mysql, Jenkins, etc.).

- Pour éviter que les reducers s'exécutent en même temps que les mappers, j'ai configuré pour que les reducers ne commencent que lorsque 99% des mappers aient complété.
- Configuration requise pour permettre à Impala de fonctionner efficacement.

## 3.2 Architecture logicielle

Dans cette section, je vais décrire comment les composantes architecturales interagissent entre elles, et expliquer le processus ETL pour arriver à la structure finale.

### 3.2.1 Architecture de Hadoop

Il serait trop long d'expliquer en détail le fonctionnement d'Hadoop, mais pour donner du contexte aux sections qui suivent, cela peut aider de présenter ses deux composantes principales.

#### 3.2.1.1 Architecture de HDFS

HDFS est le système de fichiers distribués utilisé par Hadoop. Il est composé d'un serveur de noms (NameNode) qui connaît où se trouvent tous les fichiers, et de serveurs de données (DataNode) qui transfère les fichiers directement au client.

### 3.2.1.2 Architecture de Map-Reduce

MapReduce est une plateforme permettant d'exécuter des requêtes en parallèle, sans que le programmeur ait besoin d'implémenter la gestion des communications entre les noeuds. Le client soumet une requête au JobTracker, qui à son tour distribue le travail aux TaskTracker installés sur les DataNodes.

### 3.2.2 Architecture de Hive

Hive reçoit une requête SQL d'un client et, d'une façon transparente, exprime la requête sous forme de job Map-Reduce au JobTracker. Puis Hive retourne le résultat au client, sans que le client ait à apprendre le Map-Reduce.

### 3.2.3 Architecture d'Impala

L'architecture d'Impala mérite qu'on s'y attarde plus longtemps.

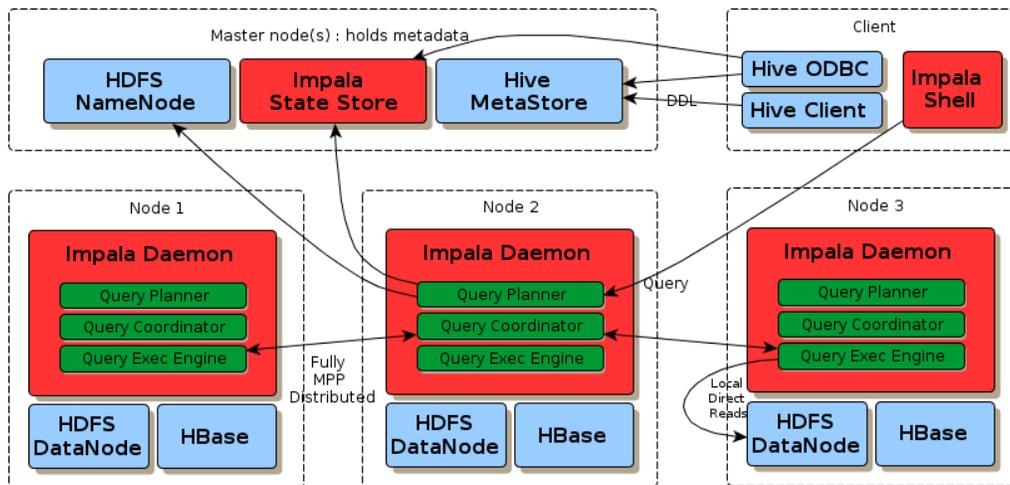


FIGURE 3.1 – Architecture d'Impala

Sur la figure 3.1, on remarque qu'Impala est étroitement reliée à Hive.

En effet, Impala est presque entièrement compatible avec le langage SQL de Hive. Impala se connecte au "metastore" de Hive pour obtenir la liste des tables disponibles. Toutefois pour l'instant, on doit passer par Hive pour créer, modifier ou supprimer la structure des tables.

Contrairement à Hive, Impala n'utilise pas map-reduce, réduisant grandement les coûts au démarrage. Plutôt il y'a un IMPalaDaemon qui doit être installé pour chaque DataNode. Impala court-circuite le chemin régulier d'accès aux données, en lisant directement les "blocks" HDFS depuis le disque.

Un client peut lancer une requête en se connectant à n'importe quel "daemon" impala. Par la suite la requête distribuée et exécutée en parallèle.

### 3.2.4 Importation des données

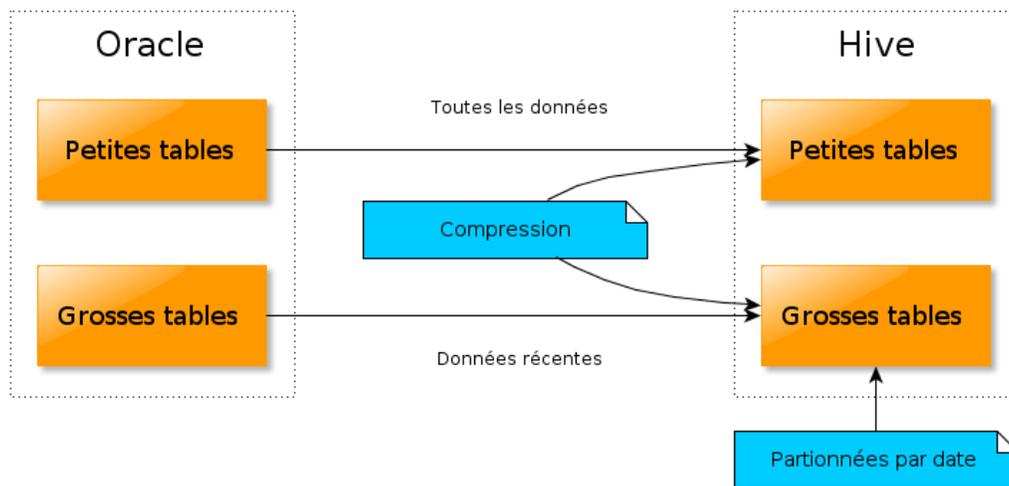


FIGURE 3.2 – Importation des tables dans Hadoop

Au niveau de l'importation des données, j'ai utilisé Sqoop pour lire les tables d'Oracle et les importer dans Hive. Il est possible d'utiliser la compression directement à cette étape, mais je ne l'ai pas utilisé. Il a fallu télécharger

le bon pilote JDBC pour correspondre à la version du serveur Oracle installé au JGH, et copier le .jar dans le dossier des librairies de Sqoop.

À partir de maintenant, je vais désigner toutes les tables ayant le préfixe "vs\_" comme étant les "petites tables", et celles avec le préfixe "vp\_" comme étant les "grosses tables". Tout est relatif à la taille dont elles occupent sur le disque.

Considérant que les petites tables peuvent changer facilement, mieux vaut importer toutes les données, à chaque synchronisation.

Pour les grosses tables, j'ai importé une année à la fois.

#### 3.2.4.1 Importation avec Quest Oracle Data Connector

Il existe un outil qui prétend améliorer les performances de Sqoop d'environ 4 fois en utilisant des méthodes Oracle plus bas niveau, mais nous n'avons pas réussi à le fonctionner. Pour être honnête, on laisse tomber rapidement étant donné que le but du projet n'était pas de faire la synchronisation. Je soupçonne que le problème est au niveau des permissions Oracle requises. Voir (Quest, c) et (Quest, a).

#### 3.2.5 Compression

Au niveau de la compression, nous avons essayé les algorithmes plus utilisés dans le domaine : GZip, Snappy. Nous avons retenu celui qui offrait le meilleur rendement **vitesse de décompression / taille compressée**; c.-à-d. Snappy.

LZO est également très utilisé avec Hive, mais au moment de faire nos tests, il n'était pas supporté par Impala.

Les références consultées incluent (Encode.ru) (Hall, 2011).

Plus de détails dans la section sur les performances (3.4).

### 3.2.6 Partitionnement

Même si Hive supporte maintenant les index, ils ne sont pas encore supportés dans Impala (toutefois, cela est prévu pour la version 1.0). Et puisque toutes les données de la table doivent être lues, il devient utile de fragmenter la table en plusieurs partitions. Si une partition est spécifiée dans la requête, alors Hive ou Impala n'a qu'à lire les données de la partition, au lieu de lire toutes les données de la table.

Comme il a été déterminé que le Dr. Eintracht utilise des dates comme critères de recherche dans 95% de ses requêtes (habituellement entre 2 semaines et 3 mois de données), il semblait évident d'utiliser la date de la commande (`ordered_date`) comme date de partition. Comme Hadoop ne gère pas bien une grande quantité de petits fichiers, j'ai créé une partition par mois plutôt qu'une partition par jour. Cela nous assure que la taille totale est plus grande que la taille d'un block HDFS.

Toutefois, il arrive que le Dr. Eintracht veuille récolter toutes les données pour un test (TestID) en particulier sans spécifier de dates. Pour éviter que Hive ait besoin de lire toute la table pour ce cas d'utilisation, j'ai créé un deuxième système de partition. Cette fois en utilisant l'année pour le premier niveau et la première lettre du TestID (abc).

La figure 3.3 explique comment ce processus est fait. J'ai utilisé Hive (au lieu d'Impala) car il permet de compresser le résultat de ses requêtes.

1. Comme les grosse tables ont en commun le `order_aa_id` et pas la `ordered_date`. La première étape consiste à extraire le lien `ordered_date / order_aa_id` dans une table séparée (pour que les jointures soient plus performantes). Cette table s'appelle "order\_dates".
2. Ensuite on fusionne les nouveaux `vp_lab_test_results` avec les `order_dates`

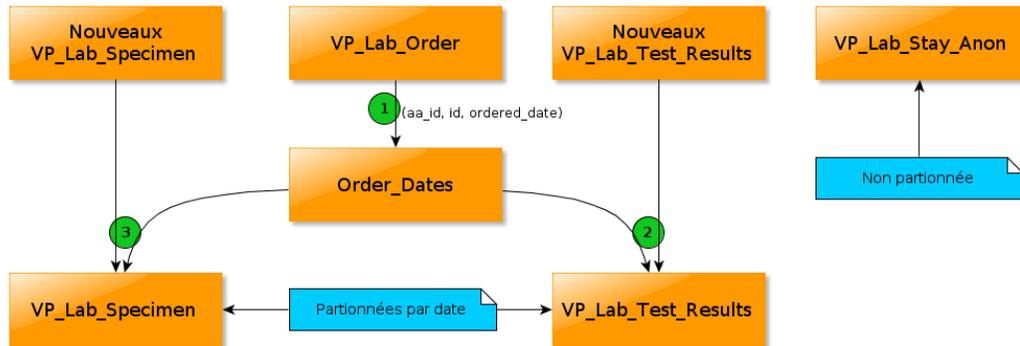


FIGURE 3.3 – Partitionnement des tables dans Hive

à l'aide d'un SQL JOIN.

- Et enfin on répète le même processus avec la table `vp_lab_specimen`.

Comme la table `vp_lab_stay_anon` est petite, il n'est pas nécessaire de la partitionner.

### 3.2.7 Dé-normalisation

Les systèmes comme Hive et Impala peuvent faire des joins, mais ils ne sont pas aussi efficaces que les RDBMS. On peut donc gagner à dénormaliser les tables, c.-à-d. de les fusionner d'avance. Encore une fois, nous avons utilisé Hive pour cette étape, car Impala ne supporte pas encore de faire des jointures sur une quantité de données qui débordent de la RAM (cette fonctionnalité est prévue rapidement après la version 1.0).

La figure 3.4 exprime toutes les étapes de cette dénormalisation :

- D'abord on fusionne `vp_lab_stay_anon`, `vp_lab_order`, et `vp_lab_test_results` ensemble. Et on ajoute des préfixes à chacun de leurs champs pour éviter les collisions (ex : deux champs ayant le même nom dans deux tables différentes). Cette première table (`stay_order_results_ym`) est partitionnée

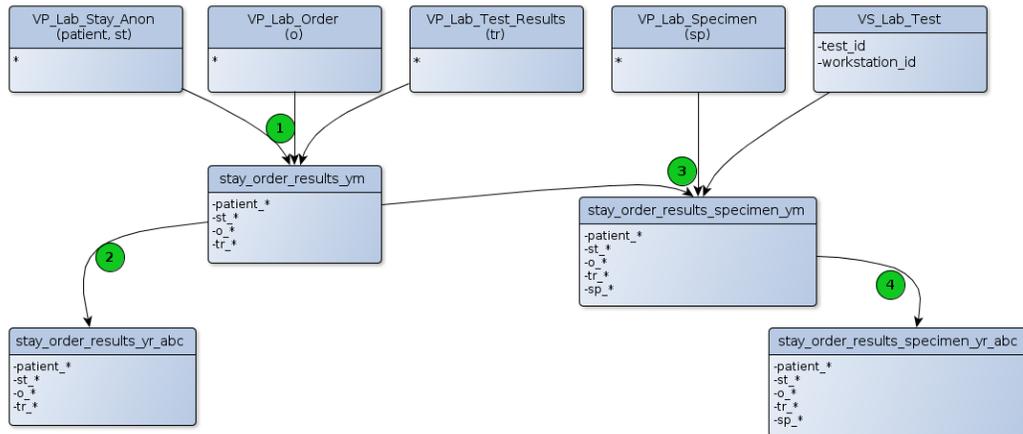


FIGURE 3.4 – Dénormalisation des tables dans Hive

par année-mois (ex. : 201203 pour mars 2012).

2. Ensuite, à partir de la table `stay_order_results_ym`, on crée une nouvelle table (`stay_order_results_yr_abc`) avec les mêmes données, mais partitionnée différemment : par année (yr), et par la première lettre du TestID (abc). Si le premier caractère n'est pas un chiffre, on remplace par '0'.
3. Après, on fusionne la table `stay_order_results_ym` avec la table `vp_lab_specimen` et la table `vs_lab_test`. La table `vs_lab_test` est utile, car il n'existe pas de lien direct entre `vp_lab_specimen` et `vp_lab_test_results`. Ce lien est douteux et certaines données sont perdues; toutefois, il s'agit de la façon dont le client travaille actuellement. Plus de détails à la section recommandation. La table créée s'appelle `stay_order_results_specimen_ym` et est partitionnée par année.
4. Similairement à l'étape 2, on crée la table `stay_order_results_specimen_yr_abc` qui est partitionnée par année (yr), et par la première lettre du TestID (abc).

Donc en résumé, le Dr. Eintracht peut utiliser n'importe laquelle des 4 tables qui s'adapte le mieux à son besoin :

- stay\_order\_results\_ym
- stay\_order\_results\_yr\_abc
- stay\_order\_results\_specimen\_ym
- stay\_order\_results\_specimen\_yr\_abc

### 3.2.8 Coordination des opérations

Pour la coordination des opérations, nous avons utilisé un outil d'intégration continue qui s'appelle Jenkins. Nous pouvons paramétrer Jenkins pour exécuter des scripts bash. Par exemple, tout ce que nous avons fait dans le projet se trouve dans Jenkins, et cela inclut les étapes de transformation des données mentionnées dans les sections précédentes. Il reste un nettoyage à faire, mais tout le code exécuté est là.

Jenkins est utile par exemple pour exécuter à distance des jobs qui nécessitent beaucoup de temps (par ex : conversion des données). On peut créer des jobs, et chaque job peut exécuter plusieurs scripts. Jenkins garde en mémoire les "logs" d'exécution, ce qui permet de facilement comparer l'exécution de plusieurs jobs.

Pour accéder à Jenkins, il suffit d'ouvrir un navigateur sur le serveur d'Endomine et le port 8080. La figure 3.5 montre l'interface principale de Jenkins. On peut voir dans le "Build Executor Status" la liste des opérations que Jenkins effectue. Dans la section de droite, les onglets représentent des vues sur une liste de jobs. La liste des jobs est en dessous des onglets.

Pour voir les détails d'une job, il suffit de cliquer sur son nom, et c'est ce qui nous amène à la vue d'une job (Figure 3.6). Dans cette vue, on peut voir

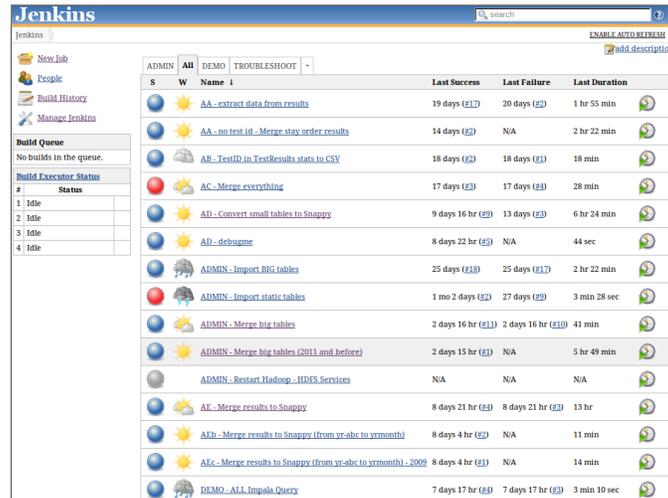


FIGURE 3.5 – Interface principale de Jenkins

à gauche l'historique des exécutions ("Build History").

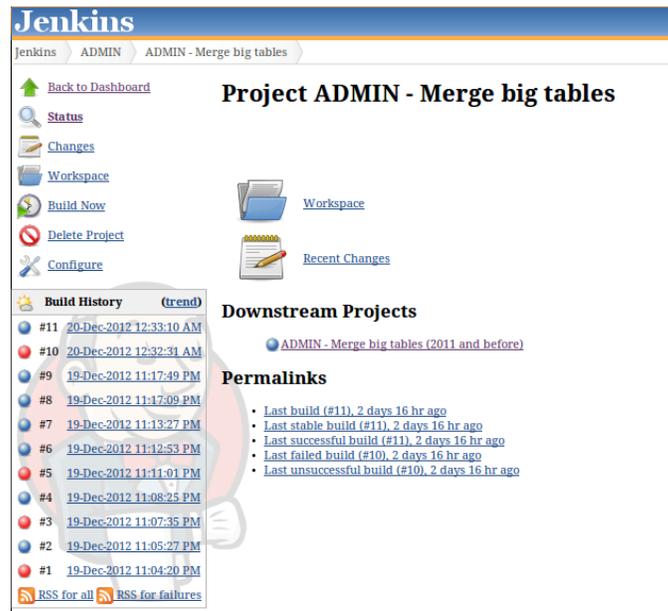


FIGURE 3.6 – Vue d'une job de Jenkins

En cliquant sur l'hyperline "Configure", on peut accéder à la liste des scripts bash exécutés par la job (voir Figure 3.7).

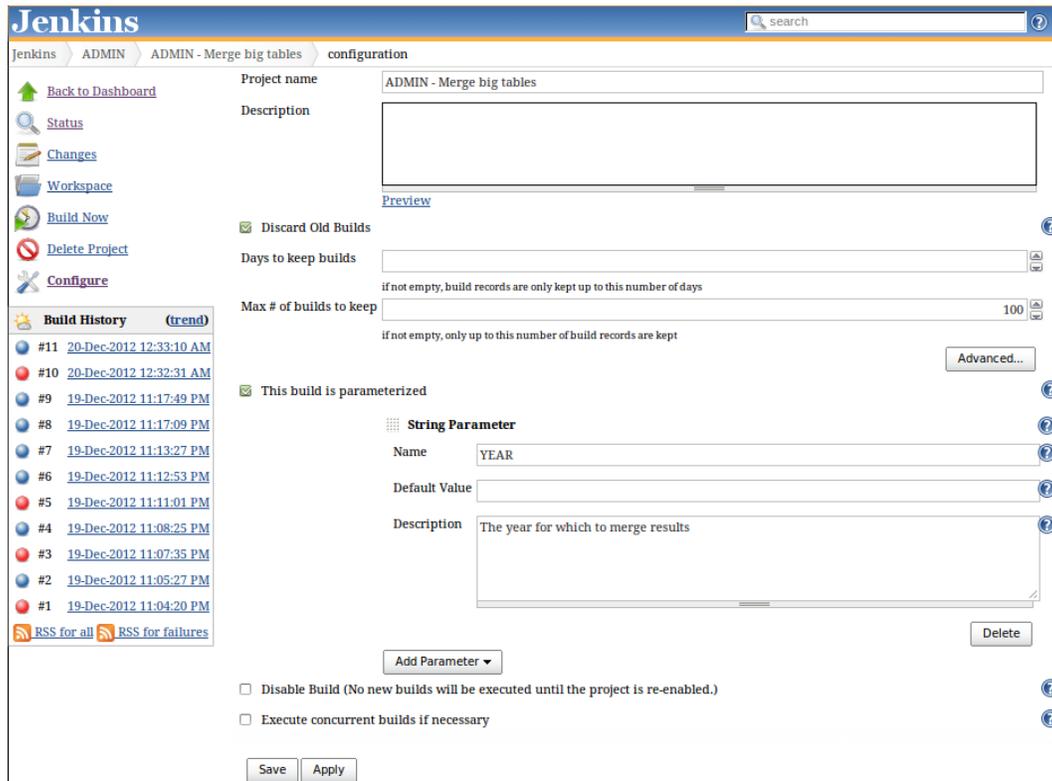


FIGURE 3.7 – Configuration d’une job de Jenkins

### 3.3 Problèmes rencontrés

Quelques problèmes qui sont survenus lors de ce projet :

- Importation des données depuis Oracle. Du au grand volume de données (320 millions de rangées pour la table des résultats), nous avons du essayé plusieurs paramètres de Sqoop. Par exemple, il fallait écrire le nom des tables Oracle en majuscule; mais la cause de ce problème a pris du temps à être identifiée, car le message d'erreur n'était pas évident.
- Installation d'Impala. Il s'agit d'un logiciel bêta et il n'y avait pas d'instructions de compilation. Parfois lorsqu'Impala plantait, le message d'erreur n'est pas explicite non plus.
- Manque de RAM forçant le système d'exploitation à "swapper". Au départ le système n'avait que 4 GB de RAM, et considérant que les services Hadoop et autres prennent pratiquement 3GB, il ne restait plus grand-chose pour effectuer des requêtes et charger les données en mémoire. La figure 3.8 illustre ce problème. On y voit que le CPU est en pratiquement en vacance (70% idle), et 28% en attente après le IO du disque (*waiting*). J'ai changé la swappiness du kernel Linux pour qu'il swap au minimum, mais après avoir vidé les *buffers* et la *cached*, le OS n'a juste pas le choix de swapper. Ce problème a été résolu avec l'ajout de 12GB de RAM.
- Espace temporaire pour map reduce. Les jobs de hive plantaient en disant qu'elles manquaient d'espace disque, mais lorsque je regardais (après le nettoyage de la job), il restait plusieurs GB d'espace disque disponible. Bien sur il s'agissait que la job hive remplît le disque, plantait et ensuite effaçait les fichiers temporaires.
- Manque de mémoire pour les map reduce. J'ai réglé ce problème en effectuant les configurations mentionnées à la section 3.1.4.

- Et j'en passe... Dans la plupart des cas les problèmes pouvaient se résoudre en ajustant des paramètres de configuration.

Plusieurs de ces problèmes sont documentés en détail, à l'annexe F.

```
top - 01:33:23 up 7:35, 8 users, load average: 2.12, 2.15, 2.16
Tasks: 224 total, 1 running, 223 sleeping, 0 stopped, 0 zombie
Cpu(s): 0.7%us, 0.4%sy, 0.0%ni, 70.7%id, 28.1%wa 0.0%hi, 0.0%si, 0.0%st
Mem: 3918724k total, 3810628k used, 108096k free, 552k buffers
Swap: 3906552k total, 3558908k used, 347644k free, 15096k cached
PID to kill: 

```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
9124	endomine	20	0	7165m	2.5g	2452	S	2.7	66.6	1:48.22	impalad
58	root	20	0	0	0	0	S	0.7	0.0	1:41.61	kswapd0
7541	mapred	20	0	1848m	76m	1648	S	0.7	2.0	2:14.81	java
17447	root	20	0	15220	852	508	R	0.7	0.0	0:27.82	top
7253	hdfs	20	0	1720m	43m	1684	S	0.3	1.1	0:14.54	java
7418	mapred	20	0	1851m	66m	1556	S	0.3	1.7	0:44.09	java
9241	endomine	20	0	170m	1672	704	S	0.3	0.0	0:34.73	python
1	root	20	0	19400	392	300	S	0.0	0.0	0:00.76	init
2	root	20	0	0	0	0	S	0.0	0.0	0:00.01	kthreadd

FIGURE 3.8 – Preuve que le système "swappait"

## 3.4 Performance

Avec Impala on arrive à lire toutes les données de 2012 (`stay_order_results_specimen_ym`) en moins d'une minute. Plus de détails sur les performances sont disponibles à la section 5.

## 3.5 Interface visuelle

L'interface visuelle prévue est décrite dans le SRS à la section 7 intitulée "Interfaces". À l'heure d'écriture de ce rapport, nous n'avons pas terminé son implémentation. Il suffirait probablement de quelques jours pour compléter cette étape.

# Chapitre 4

## Phase II-b : Prototype Oracle

Cette section décrit le prototype basé sur la base de données Oracle. L'architecture sélectionnée est relationnelle, car le modèle multidimensionnel ne permettait pas de répondre aux besoins du client.

### 4.1 Installation

Oracle est déjà installé chez le client. Pour installer chez soi, il faut télécharger la version Entreprise d'Oracle. Une fois installé, changer le fichier **ORACLE\_HOME/NETWORK/ADMIN/listener.ora** pour qu'il contienne votre adresse ip. Dans le cas que la variable d'environnement **ORACLE\_HOME** n'existe pas, la créer et la faire pointer vers le dossier d'installation d'oracle. Une fois Oracle installé, créer un **BIGFILE TABLESPACE**. Voir le fichier **grant.sql**, pour savoir comment créer le "tablespace", ainsi que les droits nécessaires à l'utilisateur.

## 4.2 Architecture logicielle

Les tables utilisées sont décrites dans la Figure : 1.4. Les sous-sections qui suivent décrivent les différentes stratégies utilisées pour augmenter les performances de notre système Oracle.

### 4.2.1 Denormalisation

Une des solutions envisagées était de fusionner les tables `v_p_lab_pat_stay_anon`, `v_p_lab_order`, `v_p_lab_specimen`, `v_p_lab_test_result` dans une même grande table. Le problème avec cette solution est que nous n'utiliserions que très peu de fonctionnalités d'Oracle, car nous serions obligés de faire des "full table scans" la majorité du temps. Ceci est, car il y a beaucoup de champs qui sont utilisés comme critères, mais sur lesquels on ne peut pas faire d'index. La raison de ne pas pouvoir faire les index est qu'ils sont très gros en mémoire (environs 5.5G pour une table comme `v_p_lab_test_results`) et qu'ils ne sont plus utilisés une fois que le nombre de rangées sélectionnées dépasse le million de valeurs (pour la table `v_p_lab_test_result`).

Ceci se produirait souvent puis que la majeure partie du temps il y a plus qu'un critère de sélection, et que prit séparément chaque critère sélectionnerait plus d'un million de rangées. Une autre raison est que `v_p_lab_specimen` est rarement utilisé avec la table `v_p_lab_test_result`. Finalement joindre les trois tables en faisant une requête "ad hoc", prends seulement 1 ordre de magnitude de temps de plus.

```

1 — 1 table 19s
2 select SUM(AA\_ID) ,COUNT(*) from V\_P\_LAB\_TEST\_RESULT where TEST\_ID='ALKI
   ' ;
3 — join the 3 tables: 136 s
4 select SUM(r.AA\_ID) ,SUM(o.ordered\_date) ,SUM(s.aa\_id) , COUNT(*) from V\_P\_
   _LAB\_TEST\_RESULT r , v\_p\_lab\_order o , v\_p\_lab\_specimen s

```

```
5 where o.AA_ID = s.order_aa_id and r.order_aa_id = o.AA_ID and TEST_ID=
    'ALKI';
```

Par contre, la normalisation est toute à fait adapté pour les solutions de "BigData".

### 4.2.2 Clustering

Considérant la nature des tables `v_p_lab_pat_stay_anon`, `v_p_lab_order`, `v_p_lab_specimen`, un "cluster" s'avérait naturel (stay contient order qui contient specimen). Nous avons essayé, mais nous n'avons pas réussi à faire un cluster. Nous décrivons la raison dans la section 4.3.

Pour le code voir `create_cluster_test.sql`.

### 4.2.3 Partitionnement

Étant donné que les tables , `v_p_lab_order`, `v_p_lab_specimen`, `v_p_lab_test_result`, sont tous joints sous la même clé (`ORDER_AA_ID`), nous avons partitionné ces tables selon cette clé.

Voici les raisons qui nous ont poussés vers la technique du partitionnement :

- Nous avons voulu initialement créer un cluster avec les tables `v_p_lab_order`, `v_p_lab_specimen` et `v_p_lab_pat_stay_anon`, mais nous n'avons pas réussi. Voir la section des problèmes rencontrés.
- Les partitions peuvent être localisés dans différents "TABLESPACE" sur différents disques. Ceci permettrait d'utiliser le parallélisme au maximum.
- Les jointures sont très rapides, car Oracle est assez intelligent pour joindre les partitions entre elles. Ceci est important, car sinon un full

table scan est effectué sur `v_p_lab_test_result` dès que le nombre `ORDER_AA_ID` dépasse 1 million.

- La table `v_p_lab_test_result` est sous partitionnés par hachage sur la clé **TEST\_ID**. Ceci est fair puisque `TEST_ID` est souvent utilisé comme critère de recherche. De plus, ça nous évite de créer un index sur la colonne **TEST\_ID**. Ceci est important, car la taille d'un index de la table `v_p_lab_test_result` pèse au bas mot 5.5G. Ceci est 5 fois plus que la taille d'un "HASH" pour un **TEST\_ID**.

Il y a plusieurs façons d'importer les données dans les tables partitionnées. La première est de créer une table vide partitionnée, puis d'importer les données dedans avec un outil "data pump". Cette solution est la plus rapide et peut être exécutée par l'utilisateur s'il possède les droits d'export/import, mais pendant l'importation la table n'est pas accessible. Le script **tables\_partitioning.sql** illustre cette façon de faire. L'avantage de cette façon de faire est que puisque la BD miroir devrait être restaurée d'un fichier sauvegarde, on pourrait l'importer directement dans une table intermédiaire. Ceci sauverait Beacoup de temps, car l'importation de la table `v_p_lab_test_result` sans partitionnement prend environ 1 heure. On peut aussi faire un "CREATE as SELECT", mais ce serait du gaspillage, car on aurait pu importer directement la table.

Dans le cas que la table à partitionner existe déjà, et si on veut y préserver l'accès pendant la durée du transfert, on peut utiliser le mode "online availability". Nous avons testé ce mode et il prend environ 7 fois plus de temps (835s vs 139.38s pour une simple copie). Le fichier **tables\_partitioning\_online\_availability.sql** illustre comment utiliser cette solution.

## 4.2.4 Compression

Nous avons essayé la compression avec des tables compressées complètement. D'après les tests, la compression augmentent le temps nécessaire pour faire les jointures;

```
1 — Tables non compress s : 58.22s
2 create table stay\_order as select s.PATIENT\_ANON\_ID,o.*
3 from v\_p\_lab\_order o, v\_p\_lab\_pat\_stay\_anon s
4 where o.stay\_aa\_id = s.aa\_id and s.aa\_id < 100000;
5
6 — Tables compress s : 462.305s
7 create table stay\_order\_comp as select s.PATIENT\_ANON\_ID,o.*
8 from v\_p\_lab\_order\_comp o, v\_p\_lab\_pat\_stay\_anon\_comp s
9 where o.stay\_aa\_id = s.aa\_id and s.aa\_id < 100000;
```

## 4.3 Problèmes rencontrés

Vu une grande quantité des problèmes rencontrés, je vais lister ceux qui sont les plus importants.

1. **Oracle 11g XE, a une limite d'espace de 11Gb.** Nous avons dû installer "Oracle Enterprise Edition" à cause de ça.
2. **L'encodage n'est pas le même entre les bases de données.** Les champs de type Varchar2 prennent de l'espace différent dépendant de l'encodage. Pour cette raison quand on copie les données d'une base de données en LATIN1 vers UTF8, des champs de type Varchar2 peuvent déborder.
3. **Le "SMALLFILE TABLESPACE" a une taille de 32Gb sous les conditions normales avec une 8Kb par block.** On a dû créer un "BIGFILE TABLESPACE" pour pouvoir sauvegarder la BD au complet.

Même s'il est possible d'avoir des multiples "SMALL TABLESPACE", il est plus simple d'en créer une grosse.

4. **Partitionner une table prends trop de temps.** Il est très important de spécifier une bonne taille initiale pour les partitions ainsi que les étendues de taille raisonnables. Sinon l'action d'étendre les partitions prend énormément de temps. (Probablement moins avec un "SMALL TABLESPACE", puisqu'il est composé d'une multitude de petits fichiers).
5. **Ça prends trop de temps de créer un "cluster".** Il faut spécifier la taille de chaque partie du cluster. Si la taille est trop petite, les étendues sont faites à plus finir. Sinon il y a un grand gaspillage d'espace. Nous avons essayé aux milieux, mais après plus de 2 heures d'attente pour des tables mesurant chacune moins de 1Gb nous avons abandonné. Voir le script `create_cluster_test.sql` pour tester.

## 4.4 Performance

Sélectionner 1 million de test\_results avec TEST\_ID quelconque prend environ 30s. Comme le "hashing" n'est pas équivalent, ça dépend vraiment des sous-partitions. De plus comme le laptop sur lesquels oracle est installé n'est pas très puissant, nous n'avons pas vue une vitesse de lecture de plus de 30 Mbs par seconde. Mais cette vitesse n'est pas constante. Par exemple :

```

1 select count(*) from v\p\lab\_test\_result r where r.TEST\_ID='PLAT'; -- 71 s
   - 3.7 Millions de rang es
2 select count(*) from v\p\lab\_test\_result r where r.TEST\_ID='EOS%'; -- 57
   s - 3.7 Millions de rang es
3 select count(*) from v\p\lab\_test\_result r where r.TEST\_ID='AUER'; -- 24
   s - 250K rang es

```

Le même principe s'applique aux jointures. Par contre, étant donné que le laptop utilisé a très peu de RAM les situations plus complexes font intervenir des "PAGE FAULTS", ce qui ralentit considérablement les requêtes.

Le temps d'importation de toutes les données, plus la création des indexes, prends environs 2h30.

## 4.5 Interface visuelle

Le générateur de requêtes "Microsoft Access" peut être utilisé comme interface graphique pour créer les requêtes sur Oracle. Dr. Shawn est déjà familier avec cette interface.

# Chapitre 5

## Comparaison des deux solutions

Dans cette section nous allons comparer les performances des solutions Oracle, Impala et Hive ainsi que l'impact des différentes options d'optimisation.

Les scripts servant à générer les différentes données se retrouvent à la fin de la section. À noter que la syntaxe SQL n'est pas exactement la même pour Hive, Oracle et Impala. Nous avons donc écrit les requêtes de façon à extraire plus ou moins les mêmes données. Tous les scripts SQL se trouvent dans "**tables/ScriptsPourComparaison.sql**".

La figure 5.1 montre la vitesse de jointure ou de copie des tables. La solution Oracle peut être jusque 2 fois plus rapide pour créer des nouvelles tables. Il ne semble pas avoir de différence en termes de jointures. Il n'y a pas de façon d'optimiser la vitesse de création de tables dans Impala.

La figure 5.2 montre l'impact de la compression dans impala, ainsi que la vitesse d'agrégation. La requête est une simple agrégation de tous les champs dans une table donnée. En général, Impala est jusqu'à 5 fois plus rapide qu'Oracle pour faire des agrégations. Les meilleurs résultats avec Impala sont obtenus en utilisant la compression snappy, ce qui permet d'augmenter la vi-

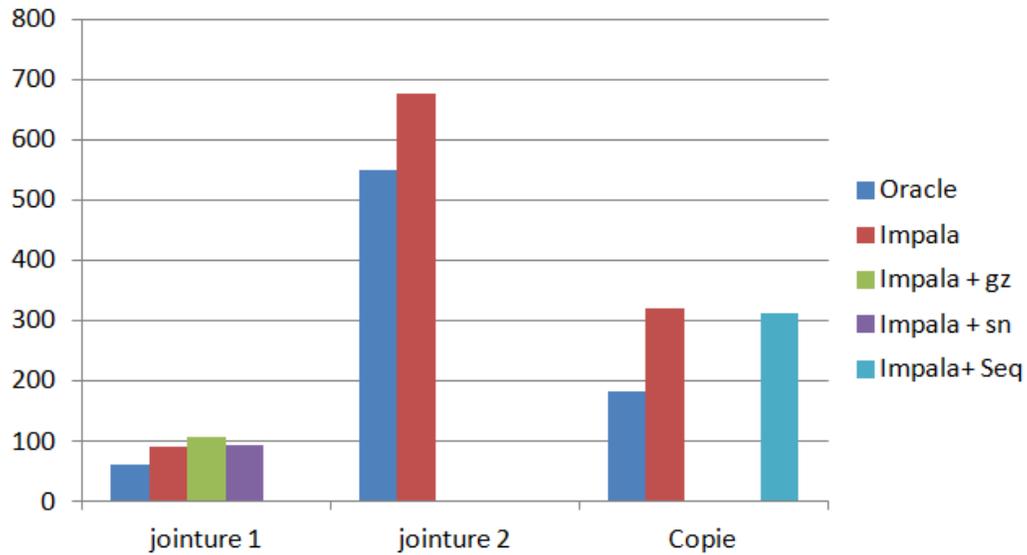


FIGURE 5.1 – Des requêtes de jointure et de copie.

tesse d'environ 25%. Hive est toujours plus lent que les deux autres solutions.

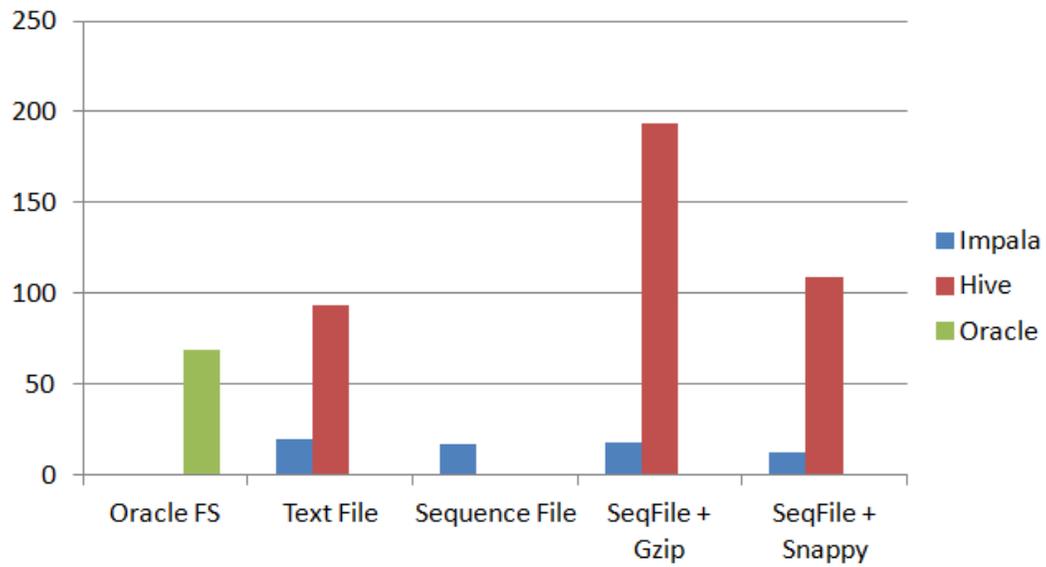


FIGURE 5.2 – Impact de compression avec différent types de fichiers.

La figure 5.3 montre le temps requis pour faire des requêtes réelles. Les

données sont divisées en trois séries :

- La série « Oracle » signifie qu'on a pris seulement la table `vp_test_results`, sans jointures. La raison est que certaines requêtes du client n'ont pas besoin de toutes les tables.
- Dans la série « Oracle Complet », on a fait la jointure de toutes les grosses tables (`stay`, `order`, `test_results`, `specimen`, et `vs_lab_test`), c.-à-d. comme Impala.
- La série Impala a utilisé la table `stay_order_results_specimen_ym`. Voir la section 3.2.7 pour plus de détails.

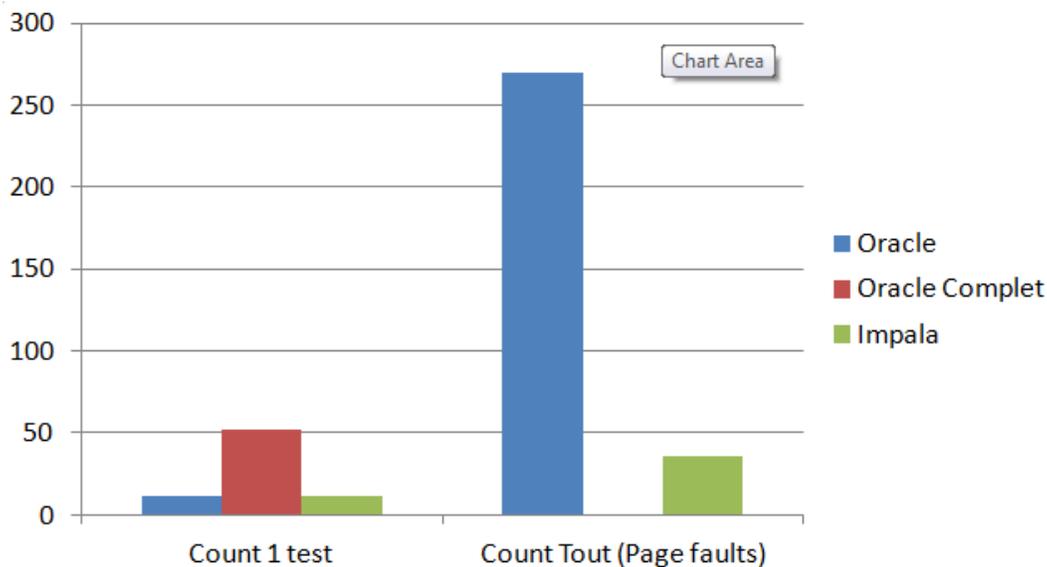


FIGURE 5.3 – Requêtes sur les données réelles.

Cette figure montre 2 tests qui ont été effectués :

- Dans le test « Count 1 test », on effectuée une requête `COUNT(*)` avec le paramètre `TestID = "ALKI"` pour les années 2009-2012. On remarque que Oracle et Impala offre des performances similaires.
- Dans le test « Count Tout (Page faults) », on a tenté de faire un `COUNT(*)`

sans TestID pour les années 2009-2012. Pour Impala, cela a fonctionné. Mais « Oracle » générait trop de "page faults" (c.-à-d. que Windows "swappait"), alors on a gardé le paramètre TestID pour « Oracle », mais même avec ce changement la solution « Oracle » devient jusqu'à 6 fois plus lente que Impala lorsqu'il y a un grand nombre de données. Étant donné les performances de « Oracle », le test n'a pas été poursuivi pour « Oracle complet ». Ce test démontre clairement que si la machine manque de mémoire vive, les performances de la solution Oracle se détériorent très rapidement.

# Conclusion

Dans ce travail nous avons analysé les besoins en termes de forage de données des clients au JGH. Afin de limiter la portée du projet, nous avons décidé d'optimiser la base de données que Dr. Eintracht utilise pour faire du forage de données.

Après avoir analysé différentes solutions "BigData", nous avons déterminé que la solution Impala/Hive semble répondre le mieux aux besoins du Dr. Eintracht à long terme.

Nous avons créé 2 prototypes, un en Oracle et l'autre Impala et comparé les solutions. Bien que les machines de test soient différentes, nous avons pu tirer plusieurs conclusions. La solution Oracle est plus flexible et plus facile à maintenir (due à la dénormalisation moindre), que la solution Impala. Mais ceci impacte les performances d'Oracle sur des grosses requêtes (avec plusieurs jointures). Dans le cas que le nombre de données recherchées est petit, les deux solutions devraient offrir des performances similaires. Il faut par contre faire très attention, car les performances d'Oracle dégradent grandement si la quantité de mémoire vive est insuffisante. La mise à l'échelle de la solution Oracle coûterait, beaucoup plus cher, que de la solution Impala. Finalement, la solution Oracle est plus facile à maintenir due à l'expertise déjà présente dans l'hôpital.

# Recommandations

Durant les tests effectués, la lecture / écriture du processus sur oracle était de l'ordre 2-30 MB/s, alors que la solution Impala sur 3 disques pouvait aller jusqu'à 200 MB/s. La raison est probablement due à la mémoire limitée sur la machine Oracle, ainsi que le nombre de "seek" sur le disque dur nécessaires pour compenser. Il faut aussi mentionner que la machine Impala possède 3 disques pour ses données contre 1 seul pour Oracle. Considérant la taille limitée des données du Dr. Eintracht, utiliser un disque SSD avec une machine ayant au moins 16GB de RAM, pourrait augmenter la vitesse de 1 ou 2 ordres de Magnitude.

La solution Oracle est plus stable, plus facile à maintenir et à une interface utilisateur déjà incluse. Nous proposons donc, s'il est possible d'avoir une licence sur une machine de test, d'implémenter cette solution pour le Dr. Eintracht, jusqu'à ce qu'une solution stable d'Impala soit complétée.

Au niveau de la solution d'Impala plusieurs améliorations pourraient être effectuées :

1. Récolter la rétroaction du Dr. Eintracht et effectuer les modifications de l'interface graphique. Par exemple, un outil graphique pourrait lui présenter la quantité de données qui sera recherchée afin de lui permettre d'estimer le temps que prendrait une requête. Pour ce faire, il suffirait

de lister la taille des fichiers contenus dans les partitions visées par la recherche.

2. Il serait possible d'investiguer l'utilisation de Quest Data Data Connector avec Sqoop pour améliorer les performances d'importation.
3. Aussi en ce moment, les partitions sont créées par année, par année-mois et par TestID. Il faudrait exécuter plus de test avec le Dr. Eintracht pour vérifier s'il ne serait pas possible d'améliorer l'organisation des partitions (ex : par semaine, ou grouper ensemble des tests qui se font souvent recherchés dans la même requête). Toutefois il faut faire attention à ne pas créer trop de partition, sinon il y aura trop de petits fichiers; ce qui rend HDFS inefficace.
4. Il faudrait faire des backups des fichiers d'image HDFS du NameNode au moins une fois par semaine, car il n'y a pas de backup pour le NameNode dans l'organisation actuelle. Il faudrait aussi ajouter plus de noeuds pour installer le SecondaryNameNode sur une machine distante du NameNode. Aussi en ajoutant plus de noeuds, cela améliorerait les performances de lectures, car on pourrait augmenter le facteur de réplique à 3.

# Références

Akbar Ahmed. Install pentaho bi server 4.5 on ubuntu 12.04 lts desktop, may 2012a. URL <http://akbarahmed.com/2012/05/24/install-pentaho-bi-server-4-5-on-ubuntu-12-04-lts-desktop/>. Consulté le 2012-10-19.

Akbar Ahmed. Install kettle 4.3.0 on ubuntu 12.04 lts, may 2012b. URL <http://akbarahmed.com/2012/05/29/install-kettle-4-3-0-on-ubuntu-12-04-lts/>. Consulté le 2012-10-19.

Saubhik Banerjee. Partitioning an existing non partitioned table with data; index and constraints - part i, sep 2011. URL <http://saubhane.blogspot.in/2011/09/partitioning-existing-non-partitioned.html>. Consulté le 2012-12-13.

Alex Baranau. Hbasewd : Avoid regionserver hotspotting despite sequential keys, apr 2012a. URL <http://blog.sematext.com/2012/04/09/hbasewd-avoid-regionserver-hotspotting-despite-writing-records-with-sequential-1>. Consulté le 2012-10-09.

Alex Baranau. Re : How to query by rowkey-infix, aug 2012b. URL [http:](http://)

- [//permalink.gmane.org/gmane.comp.java.hadoop.hbase.user/28109](http://permalink.gmane.org/gmane.comp.java.hadoop.hbase.user/28109).  
Consulté le 2012-10-15.
- Alex Baranau. Configuring hbase memstore : What you should know, jul 2012c. URL <http://blog.sematext.com/2012/07/16/hbase-memstore-what-you-should-know/>. Consulté le 2012-10-19.
- Oracle Base. Online table redefinition enhancements in oracle database 10g release 1. URL <http://www.oracle-base.com/articles/10g/online-table-redefinition-enhancements-10gr1.php>. Consulté le 2012-12-13.
- Becky Benton. Google groups : Cloudera forum : Unable to load com.mysql.jdbc.driver, oct 2011. URL <https://groups.google.com/a/cloudera.org/forum/?fromgroups=#!topic/sqoop-user/pwdahVfAAAc>. Consulté le 2012-10-16.
- Matteo Bertozzi. Hbase i/o : Hfile, feb 2011. URL <http://th30z.blogspot.ca/2011/02/hbase-io-hfile.html?spref=tw>. Consulté le 2012-10-15.
- bizcubed. Joining data sources performing lookups and joins in pentaho data integration, sep 2012. URL <http://www.youtube.com/watch?v=na6yRrhX5yo>. Consulté le 2012-10-19.
- Famzah's Blog. Linux cached/buffers memory, sep 2010. URL <http://blog.famzah.net/2010/09/14/linux-cached-buffers-memory/>. Consulté le 2012-12-14.
- Joseph Bui. Oracle - odbc connection using ms access error, oct 2008. URL <http://stackoverflow.com/questions/247317/>

`oracle-odbc-connection-using-ms-access-error-ora-12154`.

Consulté le 2012-11-29.

Steve Callan. Connecting oracle to ms access. URL [http://www.rampant-books.com/art\\_callan\\_connect\\_oracle\\_ms\\_access.htm](http://www.rampant-books.com/art_callan_connect_oracle_ms_access.htm).

Consulté le 2012-11-29.

Cloudera. Before you install cdh4 on a single node, a. URL <https://ccp.cloudera.com/display/CDH4DOC/Before+You+Install+CDH4+on+a+Single+Node#BeforeYouInstallCDH4onaSingleNode-SupportedOperatingSystemsforCDH4>.

Consulté le 2012-10-08.

Cloudera. Installing cdh4 on a single linux node in pseudo-distributed mode, b. URL <https://ccp.cloudera.com/display/CDH4DOC/Installing+CDH4+on+a+Single+Linux+Node+in+Pseudo-distributed+Mode#InstallingCDH4onaSingleLinuxNodeinPseudo-distributedMode-InstallingCDH4withMRv10>

Consulté le 2012-10-08.

Cloudera. Deprecated properties, c. URL <http://archive.cloudera.com/cdh4/cdh/4/hadoop/hadoop-project-dist/hadoop-common/DeprecatedProperties.html>. Consulté le 2012-10-08.

Cloudera. Cdh version and packaging information, d. URL <https://ccp.cloudera.com/display/DOC/CDH+Version+and+Packaging+Information#CDHVersionandPackagingInformation-CDHVersion4.1.0Packaging>. Consulté le 2012-10-08.

Cloudera. Cdh4 installation, e. URL <https://ccp.cloudera.com/display/CDH4DOC/CDH4+Installation>. Consulté le 2012-10-08.

Cloudera. Installing cdh4 on a single linux node in pseudo-distributed mode, f. URL <https://ccp.cloudera.com/display/CDH4DOC/Installing+CDH4+on+a+Single+Linux+Node+in+Pseudo-distributed+Mode>. Consulté le 2012-10-08.

Cloudera. Hbase installation, g. URL <https://ccp.cloudera.com/display/CDH4DOC/HBase+Installation#HBaseInstallation-InstallingHBase>. Consulté le 2012-10-08.

Cloudera. Zookeeper installation, h. URL <https://ccp.cloudera.com/display/CDH4DOC/ZooKeeper+Installation#ZooKeeperInstallation-InstallingtheZooKeeperServerPackage>. Consulté le 2012-10-08.

Cloudera. Installing the sqoop rpm or debian packages, i. URL <https://ccp.cloudera.com/display/CDH4DOC/Sqoop+Installation#SqoopInstallation-installRPM>. Consulté le 2012-10-16.

Cloudera. Scoop user guide : 7.2.11. importing data into hbase, j. URL [http://archive.cloudera.com/cdh4/cdh/4/sqoop/SqoopUserGuide.html#\\_importing\\_data\\_into\\_hbase](http://archive.cloudera.com/cdh4/cdh/4/sqoop/SqoopUserGuide.html#_importing_data_into_hbase). Consulté le 2012-10-16.

Cloudera. Cloudera impala 1.0 beta documentation, k. URL <https://ccp.cloudera.com/display/IMPALA10BETADOC/Cloudera+Impala+1.0+Beta+Documentation>. Consulté le 2012-10-28.

Cloudera. Ports used by impala, l. URL <https://ccp.cloudera.com/display/IMPALA10BETADOC/Ports+Used+by+Impala>. Consulté le 2012-10-28.

- Cloudera. Cloudera impala 1.0 beta documentation, m. URL <https://ccp.cloudera.com/display/IMPALA10BETADOC/Cloudera+Impala+1.0+Beta+Documentation>. Consulté le 2012-10-30.
- Cloudera. Cloudera impala 1.0 beta release notes, n. URL <https://ccp.cloudera.com/display/IMPALA10BETADOC/Cloudera+Impala+1.0+Beta+Release+Notes>. Consulté le 2012-10-30.
- Cloudera. Introducing cloudera impala, o. URL <https://ccp.cloudera.com/display/IMPALA10BETADOC/Introducing+Cloudera+Impala>. Consulté le 2012-10-30.
- Cloudera. Hive installation, p. URL <https://ccp.cloudera.com/display/CDH4DOC/Hive+Installation#HiveInstallation-InstallingHive>. Consulté le 2012-10-30.
- Cloudera. Automated installation of cloudera manager and cdh, q. URL <https://ccp.cloudera.com/display/FREE41DOC/Automated+Installation+of+Cloudera+Manager+and+CDH>. Consulté le 2012-10-30.
- Cloudera. Cloudera manager free edition download, r. URL <https://ccp.cloudera.com/display/SUPPORT/Cloudera+Manager+Free+Edition+Download>. Consulté le 2012-10-30.
- Cloudera. Installing impala, s. URL <https://ccp.cloudera.com/display/IMPALA10BETADOC/Installing+Impala>. Consulté le 2012-10-30.
- Cloudera. Github : Cloudera : Impala source code, t. URL <https://github.com/cloudera/impala>. Consulté le 2012-10-30.

Cloudera. Cloudera connector for microstrategy download, u. URL <https://ccp.cloudera.com/display/con/Cloudera+Connector+for+MicroStrategy+Download>. Consulté le 2012-12-01.

Cloudera. Cloudera connector for microstrategy, v. URL <https://ccp.cloudera.com/display/con/Cloudera+Connector+for+MicroStrategy>. Consulté le 2012-12-01.

Cloudera. Configuring impala to work with odbc, w. URL <https://ccp.cloudera.com/display/IMPALA10BETADOC/Configuring+Impala+to+Work+with+ODBC>. Consulté le 2012-12-01.

codek. Pentaho forums : Kettle repository - use and how to create?, nov 2008. URL <http://forums.pentaho.com/showthread.php?65955-Kettle-repository-use-and-how-to-create>. Consulté le 2012-10-19.

Burleson Consulting. Oracle 11g data compression tips for the database administrator. URL [http://www.dba-oracle.com/oracle11g/sf\\_Oracle\\_11g\\_Data\\_Compression\\_Tips\\_for\\_the\\_DBA.html#benefits](http://www.dba-oracle.com/oracle11g/sf_Oracle_11g_Data_Compression_Tips_for_the_DBA.html#benefits). Consulté le 2012-12-13.

Culvert. Culvert : A robust framework for secondary indexing. URL <https://github.com/booz-allen-hamilton/culvert>. Consulté le 2012-10-16.

DataMeer. Datameer : Enterprise : Analytics at the speed of business. URL <http://www.datameer.com/enterprise/index.html>. Consulté le 2012-10-28.

Sébastien Dejean. Installing a pentaho demo server on ubuntu 6.10 server

- edition, dec 2006. URL <http://ubuntu-pentaho.blogspot.ca/>. Consulté le 2012-10-19.
- James Dixon. Pentaho hadoop series : Big data analytics : Part 1 - 5. URL <http://www.pentaho.com/resources/videos/25/hadoop-series-part-1-big-data-architecture/>. Consulté le 2012-10-21.
- Encode.ru. In-memory benchmark with fastest lzss (quicklz; snappy) compressors. URL <http://encode.ru/threads/1266-In-memory-benchmark-with-fastest-LZSS-%28QuickLZ-Snappy%29-compressors>. Consulté le 2012-11-13.
- Glenn Engstrand. Olap versus big data, nov 2011. URL <http://datawarehouse.ittoolbox.com/groups/strategy-planning/dw-projectmanagement/olap-versus-big-data-4508739>. Consulté le 2012-10-21.
- Developer Extensions. Ext grid query builder example. URL <http://www.developerextensions.com/index.php/extjs-grid-query-builder>. Consulté le 2012-10-28.
- Apache Software Foundation. Apache hbase book, a. URL <http://archive.cloudera.com/cdh4/cdh/4/hbase/book.html>. Consulté le 2012-10-08.
- Apache Software Foundation. Hfile source code, b. URL <http://hbase.apache.org/xref/org/apache/hadoop/hbase/io/hfile/HFile.html>. Consulté le 2012-10-15.
- Apache Software Foundation. Hbase api docs : Package org.apache.hadoop.hbase.coprocessor, c. URL <http://hbase.apache.org/>

- `apidocs/org/apache/hadoop/hbase/coprocessor/package-summary.html`. Consulté le 2012-10-16.
- Apache Software Foundation. Sqoop developer's guide v1.4.1-cdh4.1.0, d. URL `http://archive.cloudera.com/cdh4/cdh/4/sqoop/SqoopDevGuide.html`. Consulté le 2012-10-16.
- Apache Software Foundation. Hbase book : 16.1. using existing zookeeper ensemble, e. URL `http://hbase.apache.org/book/zookeeper.html`. Consulté le 2012-10-16.
- Apache Software Foundation. Hive jdbc driver, f. URL `https://cwiki.apache.org/Hive/hivejdbcinterface.html`. Consulté le 2012-10-28.
- Apache Software Foundation. Apache thrift installation documentation, g. URL `http://thrift.apache.org/docs/install/`. Consulté le 2012-10-30.
- Apache Software Foundation. Apache tomcat downloads, h. URL `http://tomcat.apache.org/download-60.cgi`. Consulté le 2012-11-19.
- Apache Software Foundation. Hdfs file system shell guide (hadoop 0.20.2), feb 2010. URL `http://hadoop.apache.org/docs/r0.20.2/hdfs_shell.html`. Consulté le 2012-10-08.
- Apache Software Foundation. Hive odbc driver, feb 2011a. URL `https://cwiki.apache.org/Hive/hiveodbc.html`. Consulté le 2012-12-01.
- Apache Software Foundation. Hive jdbc driver, feb 2011b. URL `https://cwiki.apache.org/Hive/hivejdbcinterface.html`. Consulté le 2012-12-01.

Apache Software Foundation. Apache incubator wiki : Drill proposal, aug 2012a. URL <http://wiki.apache.org/incubator/DrillProposal>. Consulté le 2012-10-19.

Apache Software Foundation. Hive : Language manual, aug 2012b. URL <https://cwiki.apache.org/confluence/display/Hive/LanguageManual>. Consulté le 2012-11-13.

Apache Software Foundation. Hive : Getting started, sep 2012c. URL <https://cwiki.apache.org/confluence/display/Hive/GettingStarted>. Consulté le 2012-11-13.

The Eclipse Foundation. New and notable features within birt 3.7, i. URL <http://www.eclipse.org/birt/phoenix/project/notable3.7.php>. Consulté le 2012-10-28.

Gaojinchao. Hbase mail # user - a question storefileindexsize, may 2011. URL <http://search-hadoop.com/m/hemBv1LiN4Q1/a+question+storefileIndexSize&subj=a+question+storefileIndexSize>. Consulté le 2012-10-14.

Lars George. Hbase architecture 101 - storage, oct 2009. URL <http://www.larsgeorge.com/2009/10/hbase-architecture-101-storage.html>. Consulté le 2012-10-15.

Lars George. Hadoop world 2011 : Advanced hbase schema design, 2011. URL <http://www.slideshare.net/cloudera/hadoop-world-2011-advanced-hbase-schema-design>. Consulté le 2012-10-09.

- Lars George. Hbasecon : Hbase coprocessors, 2012a. URL <http://www.slideshare.net/cloudera/3-h-base-coprocessors-hbase-con-may-2012>. Consulté le 2012-10-15.
- Lars George. Berlin buzzwords june 2012 : Advanced hbase schema design, 2012b. URL <http://www.slideshare.net/larsgeorge/hbase-advanced-schema-design-berlin-buzzwords-june-2012>. Consulté le 2012-10-09.
- Go2linux. How to clear or drop the cache buffer pages from linux memory, jan 2011. URL <http://go2linux.garron.me/linux/2011/01/how-clear-or-drop-cache-buffer-pages-linux-memory-880>. Consulté le 2012-12-14.
- Google. Google refine. URL <http://code.google.com/p/google-refine/>. Consulté le 2012-10-12.
- Justin Grégoire. Mysql odbc 32 vs 64 bit (answer from justin grégoire), mar 2010. URL <http://stackoverflow.com/questions/2381906/mysql-odbc-32-vs-64-bit>. Consulté le 2011-10-11.
- The Linux System Administrator's Guide. Chapter 7.6. the buffer cache. URL [http://www.faqs.org/docs/linux\\_admin/buffer-cache.html](http://www.faqs.org/docs/linux_admin/buffer-cache.html). Consulté le 2012-12-14.
- Dj Walker-Morgan (h online). Apache to drill for big data in hadoop, aug 2012. URL <http://www.h-online.com/open/news/item/Apache-to-Drill-for-big-data-in-Hadoop-1671686.html>. Consulté le 2012-10-19.

- Mark Hall and John Paz. Hbase input, feb 2012. URL <http://wiki.pentaho.com/display/EAI/HBase+Input>. Consulté le 2012-10-19.
- Tommy Hall. Compressing text tables in hive, jun 2011. URL <http://www.thattommyhall.com/2011/06/01/compressing-text-tables-in-hive/>. Consulté le 2012-11-13.
- Guy Harrison. Comparing hadoop oracle loaders, oct 2011. URL <http://guyharrison.squarespace.com/blog/2011/10/6/comparing-hadoop-oracle-loaders.html>. Consulté le 2012-11-13.
- Lars Hofhansl. Scanning in hbase, jan 2012a. URL <http://hadoop-hbase.blogspot.ca/2012/01/scanning-in-hbase.html>. Consulté le 2012-10-16.
- Lars Hofhansl. Musings on secondary indexes, oct 2012b. URL <http://hadoop-hbase.blogspot.ca/2012/10/musings-on-secondary-indexes.html>. Consulté le 2012-10-16.
- Alex Holmes. How partitioning; collecting and spilling work in mapreduce, sep 2012. URL <http://grepalex.com/2012/09/24/map-partition-sort-spill/>. Consulté le 2012-10-19.
- Hypertable. Hypertable vs. hbase performance evaluation ii. URL [http://hypertable.com/why\\_hypertable/hypertable\\_vs\\_hbase\\_2/](http://hypertable.com/why_hypertable/hypertable_vs_hbase_2/). Consulté le 2012-10-16.
- IBM. Informix features and benefits. URL [http://www-01.ibm.com/software/data/informix/feature.html?S\\_CMP=rnav](http://www-01.ibm.com/software/data/informix/feature.html?S_CMP=rnav). Consulté le 2012-10-21.

Google Inc., Sergey Melnik, Andrey Gubarev, Jing Jing Long, Geoffrey Romer, Shiva Shivakumar, Matt Tolton, and Theo Vassilakis. Dremel : Interactive analysis of webscale datasets, 2010. URL [http://static.googleusercontent.com/external\\_content/untrusted\\_dlcp/research.google.com/en//pubs/archive/36632.pdf](http://static.googleusercontent.com/external_content/untrusted_dlcp/research.google.com/en//pubs/archive/36632.pdf).

Consulté le 2012-10-19.

Sachini Jayasekara. Using different reporting frameworks with wso2 business activity monitor, sep 2012. URL <http://wso2.org/library/articles/2012/09/using-different-reporting-frameworks-wso2-business-activity-monitor>.

Consulté le 2012-10-28.

Jenkins. Installing jenkins as a unix daemon, feb 2011a. URL <https://wiki.jenkins-ci.org/display/JENKINS/Installing+Jenkins+as+a+Unix+daemon>. Consulté le 2012-11-19.

Jenkins. Jenkinslinuxstartupscript, feb 2011b. URL <https://wiki.jenkins-ci.org/display/JENKINS/JenkinsLinuxStartupScript>.

Consulté le 2012-11-19.

Jenkins. Jenkins tomcat installation, sep 2011c. URL <https://wiki.jenkins-ci.org/display/JENKINS/Tomcat>. Consulté le 2012-11-19.

Jenkins. Starting and accessing jenkins, nov 2012. URL <https://wiki.jenkins-ci.org/display/JENKINS/Starting+and+Accessing+Jenkins>.

Consulté le 2012-11-19.

jfarrell. Thrift library for python. URL <http://people.apache.org/~jfarrell/thrift/0.7.0/lib/python/>. Consulté le 2012-10-30.

- Jimbojw. Understanding hbase and bigtable, may 2008. URL [http://jimbojw.com/wiki/index.php?title=Understanding\\_Hbase\\_and\\_BigTable](http://jimbojw.com/wiki/index.php?title=Understanding_Hbase_and_BigTable). Consulté le 2012-10-16.
- Sreejith P. K. Hbase mail # user - stargate+hbase, mar 2011. URL <http://search-hadoop.com/m/nvbiBp2TDP/Stargate%252Bhbase&subj=Stargate+hbase>. Consulté le 2012-10-15.
- William Kang. grokbase : [hbase-user] hbase internally row location mechanism, oct 2010. URL <http://grokbase.com/t/hbase/user/10ab7vvfzy/hbase-internally-row-location-mechanism>. Consulté le 2012-10-16.
- N Keywal. Hbase-user mailing list : Zookeeper available but no active master location found, apr 2012. URL [http://mail-archives.apache.org/mod\\_mbox/hbase-user/201204.mbox/%3CCAPcDmSviyQXUG8u5dPmarjakFddndyCbm6Pcgd8T6AR-aErw3g@gmail.gmail.com%3E](http://mail-archives.apache.org/mod_mbox/hbase-user/201204.mbox/%3CCAPcDmSviyQXUG8u5dPmarjakFddndyCbm6Pcgd8T6AR-aErw3g@gmail.gmail.com%3E). Consulté le 2012-10-16.
- khan. Stackoverflow : Hbase memstore and garbage collection, may 2012. URL <http://stackoverflow.com/questions/10596717/hbase-memstore-and-garbage-collection>. Consulté le 2012-10-15.
- Marcel Kornacker and Justin Erickson. Cloudera impala : Real-time queries in apache hadoop for real, oct 2012. URL <http://blog.cloudera.com/blog/2012/10/cloudera-impala-real-time-queries-in-apache-hadoop-for-real/>. Consulté le 2012-10-28.
- Ikai Lan. App engine datastore tip : monotonically increasing values are bad, jan 2011. URL <http://ikaisays.com/2011/01/25/>

- app-engine-datastore-tip-monotonically-increasing-values-are-bad/.  
Consulté le 2012-10-14.
- legrandlegrand. Sqleo visual query builder. URL <http://sqleo.sourceforge.net/index.html>. Consulté le 2012-10-28.
- leon. Stackoverflow : Where does hbase store all the row keys?,  
aug 2011. URL <http://stackoverflow.com/questions/6956400/where-does-hbase-store-all-the-row-keys>. Consulté le 2012-10-15.
- martinorth. Visual sql query builder. URL <http://www.cfsolutions.de/qb/>.  
Consulté le 2012-10-28.
- martinorth. Sencha : Ext.ux.window.visualsqlquerybuilder, may 2012.  
URL <http://www.sencha.com/forum/showthread.php?208444-Ext.ux.window.VisualSQLQueryBuilder>. Consulté le 2012-10-28.
- Microsoft. Managing data sources. URL [http://msdn.microsoft.com/en-us/library/windows/desktop/ms712362\(v=vs.85\).aspx](http://msdn.microsoft.com/en-us/library/windows/desktop/ms712362(v=vs.85).aspx). Consulté le 2012-12-01.
- Akash Mitra and DWBI Concepts. Understanding oracle query plan - a 10 minutes guide, nov 2012. URL <http://www.dwbiconcepts.com/database/22-database-oracle/26-oracle-query-plan-a-10-minutes-guide.html>. Consulté le 2012-11-13.
- Michael Mullany and EngineYard. 10 years of virtual machine performance (semi) demystified, oct 2009. URL <http://www.engineyard.com/blog/2009/10-years-of-virtual-machine-performance-semi-demystified/>.  
Consulté le 2012-11-05.

Ramesh Nair. Sqel.js : Lightweight javascript for building sql query strings.

URL <http://hiddentao.github.com/squel/>. Consulté le 2012-10-28.

Ramesh Natarajan and The Geek Stuff. Iptables flush : Delete / remove

all rules on redhat and centos linux, jan 2011. URL <http://www.thegeekstuff.com/2011/01/redhat-iptables-flush/>. Consulté le 2012-

11-19.

Ramesh Natarajan and The Geek Stuff. 10 oracle sqlldr command examples

(oracle sql\*loader tutorial), jun 2012. URL <http://www.thegeekstuff.com/2012/06/oracle-sqlldr/>. Consulté le 2012-11-05.

Nathan. Hbase error – assignment of -root- failure – cant connect to web

interface, sep 2011. URL <http://blog.nemccarthy.me/?p=110>. Consulté

le 2012-10-08.

Tech on the net. Oracle/plsql : Retrieve oracle version information. URL <http://www.techonthenet.com/oracle/questions/version.php>. Consulté le

2012-11-13.

OpenTSDB. Open time series database. URL <http://opentsdb.net/>.

Consulté le 2012-10-12.

Oracle. Java se 6 downloads, a. URL <http://www.oracle.com/technetwork/java/javasebusiness/downloads/>

[java-archive-downloads-javase6-419409.html](http://www.oracle.com/technetwork/java/javasebusiness/downloads/java-archive-downloads-javase6-419409.html). Consulté le 2012-

10-08.

Oracle. Using connector/odbc with microsoft access,

b. URL <http://dev.mysql.com/doc/refman/5.5/en/>

- `connector-odbc-examples-tools-with-access.html`. Consulté le 2011-10-11.
- Oracle. Configuring a connector/odbc dsn on windows, c. URL <http://dev.mysql.com/doc/refman/5.5/en/connector-odbc-configuration-dsn-windows.html>. Consulté le 2011-10-11.
- Oracle. Mysql : Download connector/odbc, d. URL <http://dev.mysql.com/downloads/connector/odbc/>. Consulté le 2011-10-11.
- Oracle. Mysql : Download connector/j, e. URL <http://www.mysql.com/downloads/connector/j/>. Consulté le 2012-10-16.
- Oracle. Migrating a microsoft access database to oracle database 11g, f. URL [http://www.oracle.com/webfolder/technetwork/tutorials/obe/db/hol08/sqldev\\_migration/msaccess/migrate\\_microsoft\\_access\\_otn.htm](http://www.oracle.com/webfolder/technetwork/tutorials/obe/db/hol08/sqldev_migration/msaccess/migrate_microsoft_access_otn.htm). Consulté le 2012-11-05.
- Oracle. Oracle big data connectors for hadoop : User guide, g. URL [http://docs.oracle.com/cd/E27101\\_01/doc.10/e27365/olh.htm](http://docs.oracle.com/cd/E27101_01/doc.10/e27365/olh.htm). Consulté le 2012-11-13.
- Oracle. Oracle big data connectors downloads, h. URL <http://www.oracle.com/technetwork/bdc/big-data-connectors/downloads/index.html>. Consulté le 2012-11-13.
- Oracle. Jdbc; sqlj; oracle jpublisher and universal connection pool (ucp) downloads, i. URL <http://www.oracle.com/technetwork/database/features/jdbc/index-091264.html>. Consulté le 2012-11-13.

- Oracle. Using subpartition templates to describe composite partitioned tables, j. URL [http://docs.oracle.com/cd/E18283\\_01/server.112/e16541/part\\_admin001.htm#i1006655](http://docs.oracle.com/cd/E18283_01/server.112/e16541/part_admin001.htm#i1006655). Consulté le 2012-12-13.
- Oracle. Oracle database vldb and partitioning guide : Creating partitions, k. URL [http://docs.oracle.com/cd/E18283\\_01/server.112/e16541/part\\_admin001.htm](http://docs.oracle.com/cd/E18283_01/server.112/e16541/part_admin001.htm). Consulté le 2012-12-13.
- Oracle. Partitioned tables and indexes, l. URL [http://docs.oracle.com/cd/B10501\\_01/server.920/a96524/c12parti.htm](http://docs.oracle.com/cd/B10501_01/server.920/a96524/c12parti.htm). Consulté le 2012-12-14.
- Stack Overflow. How do i tell if my windows server is swapping, oct 2009a. URL <http://serverfault.com/questions/74822/how-do-i-tell-if-my-windows-server-is-swapping>. Consulté le 2012-12-14.
- Stack Overflow. Clear file cache to repeat performance testing, jan 2009b. URL <http://stackoverflow.com/questions/478340/clear-file-cache-to-repeat-performance-testing/478367#478367>. Consulté le 2012-12-14.
- Panks. Stackoverflow : How to scan hbase rows efficiently, jan 2012. URL <http://stackoverflow.com/questions/8961989/how-to-scan-hbase-rows-efficiently>. Consulté le 2012-10-15.
- Rahul Patodi. Hue features, mar 2011. URL <http://www.technology-mania.com/2011/03/hue-features.html>. Consulté le 2012-10-28.
- Pentaho. Video pentaho presentation mapreduce, a. URL <http://www.pentaho.com/resources/videos/67/>

- pentaho-mapreduce-a-major-league-baseball-use-case/. Consulté le 2012-10-19.
- Pentaho. Pentaho community edition (ce) : Community wiki home, b. URL <http://wiki.pentaho.com/display/COM/Community+Wiki+Home>. Consulté le 2012-10-19.
- Pentaho. Pentaho community edition (ce) : Latest stable builds, c. URL <http://wiki.pentaho.com/display/COM/Latest+Stable+Builds>. Consulté le 2012-10-19.
- Pentaho. Configure pentaho for cloudera cdh4, d. URL <http://wiki.pentaho.com/display/BAD/Configure+Pentaho+for+Cloudera+CDH4>. Consulté le 2012-10-19.
- Quest. Quest data connector for oracle and hadoop 1.6 : User guide, a. URL <http://toadforcloud.com/servlet/KbServlet/download/4298-102-14176/oraooopuserguide.pdf>. Consulté le 2012-11-13.
- Quest. Quest data transporter for hive 1.1 : User guide, b. URL <http://toadforcloud.com/servlet/KbServlet/download/4298-102-14177/orahiveuserguide.pdf>. Consulté le 2012-11-13.
- Dell Quest. Data connector for oracle and hadoop, c. URL <http://toadforcloud.com/entry.jspa?categoryID=677&externalID=4298>. Consulté le 2012-11-13.
- Quora. What are the advantages of hadoop over distributed rdbms? URL <http://www.quora.com/What-are-the-advantages-of-Hadoop-over-distributed-RDBMS>. Consulté le 2012-10-21.

- Quora. How are bloom filters used in hbase?, feb 2011. URL <http://www.quora.com/How-are-bloom-filters-used-in-HBase>. Consulté le 2012-10-16.
- Alan Said. Hadoop + hbase + cygwin + windows 7 x64, sep 2011. URL <http://alans.se/blog/2010/hadoop-hbase-cygwin-windows-7-x64/>. Consulté le 2012-09-15.
- salk31. Redquerybuilder - javascript sql query builder ui. URL <http://redquerybuilder.appspot.com/>. Consulté le 2012-10-28.
- Servoy. Servoy : Query builder. URL <http://wiki.servoy.com/display/public/DOCS/Query+builder;jsessionid=7D0E08E4433B09432DC71E4F5584DE72>. Consulté le 2012-10-28.
- Tomer Shiran. Apache drill : Interactive analysis of large-scale datasets, aug 2012. URL <http://wiki.apache.org/incubator/DrillProposal?action=AttachFile&do=view&target=Drill+slides.pdf>. Consulté le 2012-10-19.
- J J Singh. Sqoop installation tutorial, mar 2012. URL <http://jugnu-life.blogspot.ca/2012/03/sqoop-installation-tutorial.html>. Consulté le 2012-10-16.
- Michael Stack. Hbase-1200 issue : Add bloomfilters, feb 2009. URL <https://issues.apache.org/jira/browse/HBASE-1200>. Consulté le 2012-10-15.
- Michael Stack. Hbase-3551 issue : Loaded hfile indexes occupy a good chunk of heap; look into shrinking the amount used and/or evicting unused indices, feb 2011. URL <https://issues.apache.org/jira/browse/HBASE-3551>. Consulté le 2012-10-14.

- TheNinjaSysAdmin. Setting swappiness value, jun 2011. URL <http://www.ninjasys.co.uk/linux/setting-swappiness-value/>. Consulté le 2012-10-15.
- Charles Toepfer. Install oracle java jdk 7 in ubuntu / mint 12, may 2012. URL <http://www.iokom.com/drupal/node/9>. Consulté le 2012-10-08.
- Tomer. Announcing the mapr hive odbc driver, may 2012. URL <http://www.mapr.com/blog/269?Itemid=78>. Consulté le 2012-10-28.
- Matt Tucker. Languagemanual udf, oct 2012. URL <https://cwiki.apache.org/confluence/display/Hive/LanguageManual+UDF#LanguageManualUDF-BuiltinFunctions>. Consulté le 2012-10-30.
- Ian Varleys. No relation : The mixed blessings of non-relational databases, 2009. URL [http://ianvarley.com/UT/MR/Varley\\_MastersReport\\_Full\\_2009-08-07.pdf](http://ianvarley.com/UT/MR/Varley_MastersReport_Full_2009-08-07.pdf). Consulté le 2012-10-14.
- Oracle Database VLDB and partition Guide. Partitioning concepts. URL [http://docs.oracle.com/cd/B28359\\_01/server.111/b32024/partition.htm](http://docs.oracle.com/cd/B28359_01/server.111/b32024/partition.htm). Consulté le 2012-12-01.
- Wikipedia. Pentaho, a. URL <http://en.wikipedia.org/wiki/Pentaho>. Consulté le 2012-10-19.
- Wikipedia. Classpath (java), b. URL [http://en.wikipedia.org/wiki/Classpath\\_\(Java\)](http://en.wikipedia.org/wiki/Classpath_(Java)). Consulté le 2012-10-19.
- Wikipedia. Bloom filter, oct 2012a. URL [http://en.wikipedia.org/wiki/Bloom\\_filter](http://en.wikipedia.org/wiki/Bloom_filter). Consulté le 2012-10-15.

Wikipedia. Bsd licenses, oct 2012b. URL [http://en.wikipedia.org/wiki/BSD\\_licenses#2-clause\\_license\\_.28.22Simplified\\_BSD\\_License.22\\_or\\_.22FreeBSD\\_License.22.29](http://en.wikipedia.org/wiki/BSD_licenses#2-clause_license_.28.22Simplified_BSD_License.22_or_.22FreeBSD_License.22.29). Consulté le 2012-10-30.

wikipedia. Quick benchmark : Gzip vs bzip2 vs lzma vs xz vs lz4 vs lzo, may 2012. URL [http://pokecraft.first-world.info/wiki/Quick\\_Benchmark:\\_Gzip\\_vs\\_Bzip2\\_vs\\_LZMA\\_vs\\_XZ\\_vs\\_LZ4\\_vs\\_LZ0](http://pokecraft.first-world.info/wiki/Quick_Benchmark:_Gzip_vs_Bzip2_vs_LZMA_vs_XZ_vs_LZ4_vs_LZ0). Consulté le 2012-11-25.

Dominic Williams. Quick install hbase in “pseudo distributed” mode and connect from java, jan 2010. URL <http://ria101.wordpress.com/2010/01/28/setup-hbase-in-pseudo-distributed-mode-and-connect-java-client/>. Consulté le 2012-10-08.

Cade Metz (Wired). Google’s mind-blowing big-data tool grows open source twin, aug 2012. URL [http://www.wired.com/wiredenterprise/2012/08/googles-mind-blowing-big-data-tool-grows-open-source-twin/?utm\\_source=Contextly&utm\\_medium=RelatedLinks&utm\\_campaign=Previous](http://www.wired.com/wiredenterprise/2012/08/googles-mind-blowing-big-data-tool-grows-open-source-twin/?utm_source=Contextly&utm_medium=RelatedLinks&utm_campaign=Previous). Consulté le 2012-10-19.

Jimmy Xiang and Cloudera. Cloudera developer center : Hbase write path, jun 2012. URL <http://www.cloudera.com/blog/2012/06/hbase-write-path/>. Consulté le 2012-10-19.

xianglei. phphiveadmin : Big data to drive; make easier for hive, sep 2012. URL <http://www.phphiveadmin.net/>. Consulté le 2012-10-28.

yahele. Stackoverflow : Javascript boolean search query builder interface library?, jan 2012. URL <http://stackoverflow.com/questions/9022033/>

javascript-boolean-search-query-builder-interface-library.

Consulté le 2012-10-28.

Yonatan. how to design hbase schema?, jun 2011. URL <http://stackoverflow.com/questions/375194/how-to-design-hbase-schema>.

Consulté le 2012-10-16.

Scott Henselman's Computer Zen. Vm performance checklist - before you complain that your virtual machine is slow, may 2007. URL <http://www.hanselman.com/blog/VMPerformanceChecklistBeforeYouComplainThatYourVirtualMachineIsSlow.aspx>. Consulté le 2012-11-05.

# Bibliographie

Alexander Rasin Daniel J. Abadi David J. DeWitt Samuel Madden Michael Stonebraker Andrew Pavlo, Erik Paulson. A comparison of approaches to large-scale data analysis. pages 165–178, 2009.

Kapil Bakshi. Considerations for big data : Architecture and approach. pages 1–7, 2012.

Edd Dumbill. Making sense of big data. Big Data. doi : 10.1089/big.2012.1503.

Google and A. Fikes. Storage architecture and challenges. Talk at the Google Faculty Summit, 2010a.

Google and A. Fikes. Storage architecture and challenges. Talk at the Google Faculty Summit, 2010b.

Jian Huang, Xiangyong Ouyang, Jithin Jose, Md. Wasi ur Rahman, Hao Wang, Miao Luo, Hari Subramoni, Chet Murthy, and Dhabaleswar K. Panda. High-performance design of hbase with rdma over infiniband. pages 774–785, 2012. doi : 10.1109/IPDPS.2012.74.

Zhao Jing-hua, Song Ai-mei, and Song Ai-bo. Olap aggregation based on dimension-oriented storage. pages 1932–1936, 2012. doi : 10.1109/IPDPSW.2012.241.

OlapCouncil. Olap council white paper. URL <http://www.olapcouncil.org/research/whtpapco.htm>, 1997.

Oracle. Oracle database 11g product family white paper.

Spiros Papadimitriou and Jimeng Sun. Disco : Distributed co-clustering with map-reduce : A case study towards petabyte-scale end-to-end mining. pages 512–521, 2008. doi : 10.1109/ICDM.2008.142.

Philip Russom. Tdwi checklist report : Analytic databases for big data.

TDWI. What works in emerging technologies.

# Annexe A

## Plan de travail

TABLE A.1: Échéancier

#	Début	Fin	Ef- fort Es- ti- mé	Ef- fort Ac- tuel	Phase / Itérations / Tâches / Jalon	Livrable(s) / Artéfacts	Res- pon- sa- ble(s)
	2012- 09-13	2012- 09-26	23.5	26.5	Phase d'Inception - Itération I1		
1.1	2012- 09-14	2012- 09-14	1	1	Rencontre – au JGH avec le client		DL
1.2	2012- 09-14	2012- 09-14	1.5	1.5	Rencontre – professeur superviseur		DL

TABLE A.1: (suite)

#	Début	Fin	Ef- fort Es- ti- mé	Ef- fort Ac- tuel	Phase / Itérations / Tâches / Jalon	Livrable(s) / Artéfacts	Res- pon- sa- ble(s)
1.3	2012- 09-14	2012- 09-20	1	1	Production de la fiche de renseignements	Fiche de renseigne- ments	DL
1.4	2012- 09-18	2012- 09-28	2	2	Planification du projet	Plan de projet	DL
2.1	2012- 09-20	2012- 09-20	4	4	Rencontre – au JGH avec le client		DL, AZ
2.2	2012- 09-20	2012- 09-24	4	5	Identification et répar- tition des tâches du projet par itération	Échéancier	DL
2.3	2012- 09-20	2012- 09-24	3	4	Production de la pro- position de projet	Proposition de projet	DL, AZ
	2012- 09-27	2012- 10-10	57	70	Phase d’Inception - Itération I2		
3.1	2012- 09-27	2012- 09-27	3	5	Rencontre – au JGH avec le client		DL, AZ

TABLE A.1: (suite)

#	Début	Fin	Ef- fort Es- ti- mé	Ef- fort Ac- tuel	Phase / Itérations / Tâches / Jalon	Livrable(s) / Artéfacts	Res- pon- sa- ble(s)
3.2	2012- 09-27	2012- 10-03	6	5	Analyse du système existant	Schéma relationnel de pro- duction actuel	DL, AZ
3.3	2012- 09-27	2012- 10-03	5	10	Recherche sur le forage de données à grande échelle (part 1)		DL
4.1	2012- 10-04	2012- 10-04	1.5	2	Rencontre – au JGH avec le client		DL, AZ
4.2	2012- 10-04	2012- 10-10	5	5	Identifier les pro- blèmes du système existant		DL, AZ
4.3	2012- 10-04	2012- 10-10	8	8	Identifier les besoins et caractéristiques du nouveau système, et les prioriser		DL, AZ

TABLE A.1: (suite)

#	Début	Fin	Ef- fort Es- ti- mé	Ef- fort Ac- tuel	Phase / Itérations / Tâches / Jalon	Livrable(s) / Artéfacts	Res- pon- sa- ble(s)
4.4	2012- 10-04	2012- 10-10	5	5	Recherche sur le forage de données à grande échelle (part 2)		DL
4.5	2012- 10-01	2012- 10-12	5	5	Production du do- cument de Vision (B.R.S.)	Document de Vision (B.R.S.)	AZ
	2012- 10-11	2012- 10-21	79	89	Phase d'Élaboration - Itération E1		
5.1	2012- 10-11	2012- 10-11	1.5	4	Rencontre – au JGH avec le client		DL, AZ
5.2	2012- 10-11	2012- 10-17	4	5	Concevoir les cas d'utilisations		DL, AZ
5.3	2012- 10-11	2012- 10-17	5	5	Identifier les exi- gences fonctionnelles, non-fonctionnelles, et contraintes de conception		AZ

TABLE A.1: (suite)

#	Début	Fin	Ef- fort Es- ti- mé	Ef- fort Ac- tuel	Phase / Itérations / Tâches / Jalon	Livrable(s) / Artéfacts	Res- pon- sa- ble(s)
5.4	2012- 10-11	2012- 10-17	5	5	Établir une stratégie de miroir de la BD de production actuelle		DL
5.5	2012- 10-11	2012- 10-17	5	5	Recherche sur le forage de données à grande échelle (part 3)		DL
6.1	2012- 10-18	2012- 10-18	1.5	3	Rencontre – au JGH avec le client		DL, AZ
6.2	2012- 10-18	2012- 10-21	5	5	Dénormaliser le schéma relationnel et sélection des champs importants		DL, AZ
6.3	2012- 10-18	2012- 10-21	5	5	Définir le schéma des relations Hadoop	Structure de sto- ckage des données	DL, AZ

TABLE A.1: (suite)

#	Début	Fin	Ef- fort Es- ti- mé	Ef- fort Ac- tuel	Phase / Itérations / Tâches / Jalon	Livrable(s) / Artéfacts	Res- pon- sa- ble(s)
6.4	2012- 10-18	2012- 10-21	10	10	Concevoir une archi- tecture pour le «data mining»	Schéma ar- chitectural	DL, AZ
6.5	2012- 10-18	2012- 10-21	5	5	Design des interface graphiques (recherche, résultats, etc.)		AZ
6.6	2012- 10-18	2012- 10-21	5	5	Recherche sur le forage de données à grande échelle (part 4)		DL
	2012- 10-22	2012- 11-07	95	211	Phase d'Élaboration - Itération E2		
7.1	2012- 10-22	2012- 10-22	1.5	3	Rencontre – au JGH avec le client		DL, AZ
7.2	2012- 10-22	2012- 10-22	1	2	Rencontre – professeur superviseur		DL, AZ
7.3	2012- 10-23	2012- 10-25	18	22	Formation/Conférence Hadoop World		DL, AZ





TABLE A.1: (suite)

#	Début	Fin	Ef- fort Es- ti- mé	Ef- fort Ac- tuel	Phase / Itérations / Tâches / Jalon	Livrable(s) / Artéfacts	Res- pon- sa- ble(s)
10.1	2012- 11-15	2012- 11-15	1.5	3	Rencontre – au JGH avec le client		DL, AZ
10.2	2012- 11-15	2012- 11-21	20	5	Implémentation de l'engin de recherche (part 1)		AZ
	2012- 11-22	2012- 12-05		45	Phase de Construction - Itération C2		
11.1	2012- 11-22	2012- 11-22	1.5	3	Rencontre – au JGH avec le client		DL, AZ
11.2	2012- 11-22	2012- 11-22	1	2	Rencontre – professeur superviseur		DL, AZ
11.3	2012- 11-22	2012- 11-28	5	0	Implémentation de l'engin de recherche (part 2)		AZ
11.4	2012- 11-22	2012- 11-28	5	0	Tester l'engin de re- cherche		DL



TABLE A.1: (suite)

#	Début	Fin	Ef- fort Es- ti- mé	Ef- fort Ac- tuel	Phase / Itérations / Tâches / Jalon	Livrable(s) / Artéfacts	Res- pon- sa- ble(s)
	2012- 12-06	2012- 12-14	33	55	Phase de Transition - Itération T1		
13.1	2012- 12-06	2012- 12-06	1.5	2	Rencontre – au JGH avec le client		DL, AZ
13.2	2012- 12-03	2012- 12-14	4	5	Documentation sur la continuation du projet	Spécifications de pro- chaine version	DL, AZ
13.3	2012- 12-06	2012- 12-10	2	3	Présentation oral (Da- vid Lauzon)		DL
13.4	2012- 12-06	2012- 12-10	2	3	Présentation oral (An- ton Zakharov)		AZ
13.5	2012- 12-06	2012- 12-12	4	10	Production du rapport final d'équipe	Rapport final d'équipe	DL, AZ
13.6	2012- 12-06	2012- 12-12	5	10	Production du rapport final individuel (David Lauzon)	Rapport fi- nal - DL	DL

TABLE A.1: (suite)

#	Début	Fin	Ef- fort Es- ti- mé	Ef- fort Ac- tuel	Phase / Itérations / Tâches / Jalon	Livrable(s) / Artéfacts	Res- pon- sa- ble(s)
13.7	2012- 12-06	2012- 12-12	5	10	Production du rapport final individuel (Anton Zakharov)	Rapport fi- nal - AZ	AZ
	2012- 09-13	2012- 12-14			JALON 1 : Prototype de data mining		
			208	313	Total (David)		
			185	250	Total (Anton)		
			393	563	Grand total		

# Annexe B

## Document de Vision

# ENDOMINE

PROJET DE DÉVELOPPEMENT D'OUTILS DE FORAGE DE DONNÉES DE  
RÉSULTATS DE TESTS PATIENTS - ENDOCRINOLOGIE, MÉTABOLISME ET  
ÉPIDÉMIOLOGIE CLINIQUE

Projet #2012-076

Version : 1.0.1  
Date d'émission : 4 octobre 2012  
Date de révision : 7 novembre 2012

---

## Vision

---

Présenté à :

Auteurs :



David Lauzon  
Anton Zakharov



Department of  
Diagnostic Medicine

## Table des matières

<b>1</b>	<b>Introduction</b>	<b>5</b>
1.1	Objectif . . . . .	5
1.2	Portée . . . . .	5
1.3	Définitions, acronymes et abréviations . . . . .	5
1.4	Références . . . . .	6
1.5	Langue de rédaction . . . . .	6
<b>2</b>	<b>Positionnement</b>	<b>6</b>
2.1	Énoncé du problème . . . . .	6
2.1.1	Problème 1 . . . . .	6
2.1.2	Problème 2 . . . . .	7
2.1.3	Problème 3 . . . . .	7
2.2	Positionnement du produit . . . . .	7
<b>3</b>	<b>Descriptions des intervenants et des utilisateurs</b>	<b>8</b>
3.1	Résumé des intervenants ( <i>stakeholders</i> ) . . . . .	8
3.2	Résumé des utilisateurs . . . . .	8
3.3	Environnement utilisateur . . . . .	8
3.4	Principaux besoins des intervenants et utilisateurs . . . . .	9
3.5	Alternatives et Compétition . . . . .	9
<b>4</b>	<b>Vue d'ensemble du produit</b>	<b>9</b>
4.1	Perspective du produit . . . . .	9
4.2	Sécurité et Confidentialité . . . . .	11
4.3	Principaux avantages . . . . .	11
4.4	Hypothèses et dépendances . . . . .	11
	HYP01 BD de <i>staging</i> . . . . .	11
	HYP02 Synchronisation . . . . .	11
	HYP03 Anonymisation . . . . .	12
	HYP04 Prototype . . . . .	12
	HYP05 Conception, développement, et test . . . . .	12
4.5	Licences et installation . . . . .	12
<b>5</b>	<b>Caractéristiques (<i>features</i>) du produit</b>	<b>12</b>
	FEA01 Configuration minimale pour ajouter d'autres machines. . . . .	12
	FEA02 Code supportant le parallélisme. . . . .	12
	FEA03 Distribution automatique des données pour supporter parallélisme . . . . .	12
	FEA04 Supporter plusieurs requêtes en parallèles . . . . .	13
	FEA05 Environnement de forage de données distinct des sources de données en production . . . . .	13
	FEA06 Synchronisation des données automatique entre les sources de données et le système EndoMine . . . . .	13
	FEA07 Générateur de requêtes intégré . . . . .	13
	FEA08 Recherche par filtrage incrémentiel de la requête originale . . . . .	13

FEA09	Exportation des résultats de recherche . . . . .	13
FEA10	Exportation des clés anonymisées . . . . .	13
FEA11	Ajout de champs supplémentaire à des résultats de recherche . . . . .	13
FEA12	Trier les résultats de recherche . . . . .	14
FEA13	Dés/Anonymisation des informations confidentielles . . . . .	14
FEA14	Filtrage automatique des colonnes des résultats . . . . .	14
FEA15	Extraction des valeurs uniques d'une colonne . . . . .	14
<b>6</b>	<b>Contraintes</b>	<b>14</b>
VC01	Accessibilité . . . . .	14
VC02	Confidentialité . . . . .	14
VC03	Règles du JGH . . . . .	14
VC04	Coût . . . . .	14
VC05	Modification . . . . .	14
<b>7</b>	<b>Gammes de qualité</b>	<b>15</b>
<b>8</b>	<b>Attributs des caractéristiques</b>	<b>15</b>
<b>9</b>	<b>Autres exigences du produit</b>	<b>16</b>
9.1	Exigences du système . . . . .	16
9.2	Exigences de performance . . . . .	16
<b>10</b>	<b>Exigences de documentation</b>	<b>16</b>
10.1	Manuel de l'utilisateur . . . . .	16
10.2	Guides d'installation, de configuration, et fichier à lire . . . . .	16
<b>11</b>	<b>Bibliographie</b>	<b>17</b>
<b>A</b>	<b>Microsoft Access Query Builder</b>	<b>18</b>

## Liste des tableaux

1	Historique des Révisions . . . . .	4
3	Résumé des intervenants ( <i>stakeholders</i> ) . . . . .	8
4	Résumé des utilisateurs . . . . .	8
5	Besoins . . . . .	9
6	Avantages EndoMine . . . . .	11
7	Attributs des caractéristiques . . . . .	15
8	Légende : État des caractéristiques . . . . .	19
9	Légende : Bénéfice des caractéristiques . . . . .	19
10	Légende : Effort des caractéristiques . . . . .	19
11	Légende : Risque des caractéristiques . . . . .	20
12	Légende : Stabilité des caractéristiques . . . . .	20
14	Légende : Priorité des caractéristiques . . . . .	20

## Table des figures

1	Modèle du domaine . . . . .	5
2	Perspective du produit . . . . .	10
3	Exemple d'une requête construite avec le Microsoft Access Query Builder (image tirée d'un siteweb : [2]) . . . . .	18

TABLE 1: Historique des Révisions

Date	Version	Description	Auteur
27 sept. 2012	v0.1	Sections 2.2, 2.3, 3.4, 3.7, 4.1	Anton Zakharov
27 sept. 2012	v0.2	Sections 1.3, 3.2, 3.3, 4.1, 4.3	David Lauzon
3oct. 2012	v0.2.1	Formattage et mise en page	David Lauzon
3oct. 2012	v0.3- preview	Section 1.2 et 3.7	Anton Zakharov
5oct. 2012	v0.4	Section 4.2, 4.4, 4.5, 5-7	Anton Zakharov
7oct. 2012	v0.5	Section 1.3-1.5, 3.2-3.4, 8-11	David Lauzon
8oct. 2012	v0.5.1	Revue des sections à David	Anton Zakharov
9oct. 2012	v0.9-rc1	Revue des sections à Anton	David Lauzon
11 oct. 2012	v0.9.1	Corrections suggérées par Fodil (2.1, 4.4, 4.5, 9.3).	David Lauzon
4nov. 2012	v1.0	Ajustement des descriptions des besoins. Ajout du besoin de ETL.	Anton Zakharov
7nov. 2012	v1.0.1	Changé le département et enlevé les intervenants non-pertinents	David Lauzon

# 1 Introduction

## 1.1 Objectif

Le but de ce document est de collecter, analyser, et définir les besoins et caractéristiques de haut niveau du système EndoMine. Il se concentre sur les fonctionnalités recherchées par les parties prenantes, et explique pourquoi ces besoins existent. Les documents de cas d'utilisations (UC) et les spécifications des exigences logicielles (SRS) détaillent comment EndoMine satisfait ces besoins.

## 1.2 Portée

Ce document de vision porte sur le développement et l'intégration du système de forage de données EndoMine. L'outil permettrait aux chercheurs du JGH de réaliser du forage de données à grande échelle sur la banque de données de tests biomédicaux. Les requêtes pourraient être faites à partir d'une interface utilisateur, tout en respectant les règles de sécurité et de confidentialité de l'hôpital.

La figure 1 présente les relations entre les principaux concepts dans le domaine du client. Une étoile signifie "plusieurs". Par exemple, la relation entre *Test Order* et *Test* se lit comme suis : "1 commande de tests peut comporter plusieurs tests", et la relation entre *Test* et *Test Result* se lit : "un même test n'a qu'un seul résultat". Les diagnostics (en jaune) et son lien avec le séjour n'existe pas encore dans le contexte du département de biochimie médicale, mais il est attendu qu'ils seront ajoutés en cours de projet. Se référer à la section 1.3 (Définitions, acronymes et abbréviations) pour l'explication de ces concepts et autre terminologie mentionné dans ce document.

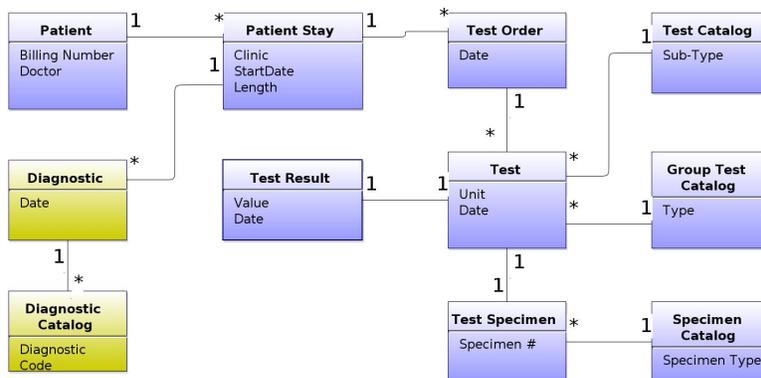


FIGURE 1 – Modèle du domaine

## 1.3 Définitions, acronymes et abbréviations

BD, DB	Base de données / banque de données / <i>database</i>
Diagnostic	Associe le séjour d'un patient à un code de diagnostic.
Diagnostic Catalog	Décrit un diagnostic (ex : diabète).
ÉTS	École de Technologie Supérieure

GÉLOG	Software Engineering Research Laboratory
Group Test Catalog	Décris le type général d'un test (ex : Testostérone).
JGH / HGJ	Jewish General Hospital / Hôpital général juif
Patient	Personne ayant un dossier (ou fiche) à l'hôpital JGH.
Patient Stay	Séjour d'un patient à l'hôpital associé à une clinique en particulier (ex : endocrinologie).
Specimen Catalog	Décris la méthode du test (ex : urine, sang, etc.)
SQL	Structured Query Language
Test	Décris le test à effectuer sur un patient. Contient des informations sur les unités du résultat de test (ex : g/mL).
Test Catalog	Décris le type spécifique d'un test (ex : Testostérone-B, Testostérone-C).
Test Order	Commande d'un ou plusieurs tests effectuée à une date précise pour un séjour en particulier.
Test Result	Contient les informations relatives au résultat du test (ex : quantité de mL de glucose).
Test Specimen	Fait référence à l'échantillon de test (ex : l'éprouvette de sang).

## 1.4 Références

Voir le Plan de Projet pour la liste des artefacts livrables de ce projet.  
Les autres références citées se trouvent à la fin dans la section 10.2 intitulée Bibliographie.

## 1.5 Langue de rédaction

La langue de rédaction principale des documents sera le français. Toutefois, lorsque approprié, l'anglais pourrait être utilisé pour s'assurer la compréhension de l'ensemble des utilisateurs d'EndoMine.

# 2 Positionnement

## 2.1 Énoncé du problème

### 2.1.1 Problème 1

Le problème de	l'extraction de statistiques complètes et précises sur l'utilisation des ressources médicales ( <i>scanner</i> , microscope, tout ce qui se trouve dans les laboratoires) est compliquée, voire impossible.
Cela affecte	les gestionnaires du laboratoire biomédical
dont l'impact est	la difficulté de prendre des décisions objectives d'achat et d'allocation de matériels.
Une bonne solution serait	d'ajouter les diagnostics médicaux et les relier aux tests médicaux dans le système de forage de données. Cela permettrait de prouver que des tests demandés par des médecins sont effectués inutilement, en vérifiant les tests avec les diagnostics.

### 2.1.2 Problème 2

Le problème de	forer des données à grande échelle temporelle (données échelonnées sur plusieurs années) prend trop de temps et ralentis le système pour tous les utilisateurs.
Cela affecte	les chercheurs et les utilisateurs d'équipements médicaux dans les laboratoires de biochimie
dont l'impact est	que les chercheurs doivent soit : a) limiter la quantité de données recherchées ou b) attendre que les résultats d'une recherche complète soient disponibles.
Une bonne solution serait	un système de forage de données rapide dont le traitement d'une recherche intensive soit quasi transparente pour les autres utilisateurs non concernés. De plus, la solution devrait s'adapter facilement à une quantité de données et à un nombre d'utilisateurs grandissants.

### 2.1.3 Problème 3

Le problème de	la limite de 1GB de résultats de MS Access ne permet pas de rechercher l'ensemble des données disponible (référence : [1]).
Cela affecte	les chercheurs utilisant MS Access
dont l'impact est	l'obligation de limiter la quantité de données traitées lors d'une même recherche.
Une bonne solution serait	un système de forage de données permettant de faire des recherches sur l'ensemble des données de SoftLab, tout en conservant une facilité d'utilisation comme MS Access.

## 2.2 Positionnement du produit

Pour	les chercheurs et gestionnaires des départements d'endocrinologie et diagnostique médicale de l'hopital général Juif
qui	veulent une solution efficace pour faire des recherches dans leur banque de données.
EndoMine	est une solution de forage de données
qui	est efficace, rapide, et permet une mise à l'échelle ( <i>scaling</i> ) à faible coût.
Contrairement à	une base données relationnelle traditionnelle, onéreuse, surchargée et lente,
notre produit	est facile à utiliser, sauve du temps précieux dans la réalisation de recherches à grande échelle, tout en conservant un faible coût de possession (TCO).

### 3 Descriptions des intervenants et des utilisateurs

#### 3.1 Résumé des intervenants (*stakeholders*)

TABLE 3 – Résumé des intervenants (*stakeholders*)

Nom	Description	Responsabilités
STK1. Dr. Elizabeth Mac Namara	Chief Medical Biochemistry (JGH)	Pilote (sponsor : states long term requirements).
STK2. Dr. Shaun Eintracht	Medical Biochemist, Dept. of Diagnostic Medecine (JGH)	Pilote (will operate and use the resulting data mining system).
STK3. Dr. Mark Trifiro	Chief Endocrinology and Metabolism (JGH)	Client (sponsor : states long term requirements).
STK5. Dr. Alain April	Directeur du GÉLOG (ÉTS)	Provide Information Technology solutions and guide ÉTS students.
STK6. Chris Polykandriotis	IT Specialist (JGH)	Help with IT issues.

#### 3.2 Résumé des utilisateurs

TABLE 4 – Résumé des utilisateurs

Nom	Description	Responsabilités	Intervenant
USR1. Utilisateur privilégié	Utilisateur ayant accès à la base de données de production Oracle, et au système EndoMine. Il s'agit des utilisateurs du département de biochimie du JGH.	<ul style="list-style-type: none"> <li>– Faire des recherches spécifiques.</li> <li>– Produire des rapports de recherche</li> <li>– Gérer l'utilisation de ressources médicales.</li> </ul>	STK1 (Dr. Mac Namara), STK2 (Dr. Eintracht)
USR2. Utilisateur restreint	Utilisateur externe au département de biochimie ayant accès seulement au système EndoMine (donc ils n'ont pas accès aux informations confidentielles des patients). Il s'agit des utilisateurs des autres départements du JGH.	<ul style="list-style-type: none"> <li>– Faire des recherches spécifiques.</li> </ul>	STK3 (Dr. Trifiro)

#### 3.3 Environnement utilisateur

- Poste de travail ayant un accès intranet au serveur d'EndoMine.

### 3.4 Principaux besoins des intervenants et utilisateurs

TABLE 5: Besoins

Besoin	Priorité	Préoccupations	Solution actuelle	Solution proposée
N1. Confidentialité des données	Critique	Informations confidentielles protégées contre l'accès de personnes non autorisées.	Mesures de sécurité inconnues.	Voir la section 4.2 ('Sécurité et Confidentialité').
N2. Forage de données n'interférant pas avec la collection de données	Critique	Comme le forage est traité directement sur l'environnement de collection des données, des pannes surviennent bloquant temporairement l'accès à tous les utilisateurs.	Aucune	Séparation distincte des environnements de production et de forage de données.
N3. Outil simple pour créer des requêtes	Critique	Courte formation pour apprendre l'outil. Aucune connaissance de SQL requise.	Dr. Eintracht utilise MS Access Query Builder	Interface similaire à MS Access Query Builder. (voir la Figure 3 de l'A)
N4. Mise à l'échelle facile du système	Important	Accomoder un nombre de données et d'utilisateurs grandissant, sans que les coûts ne deviennent exponentiels.	Incapable de répondre à ce besoin.	Extensibilité horizontale graduelle en ajoutant d'autres ordinateurs au système. La configuration à modifier serait minimale.
N5. Forage de données sans limite de taille	Important	Pouvoir rechercher toutes les données disponibles.	La mémoire de MS Access limite la taille des recherches. [1]	L'architecture d'EndoMine n'as pas cette restriction.

### 3.5 Alternatives et Compétition

A notre connaissance il n'existe pas de solution clé en main pour faire directement du forage de données sur la structure de la base de données actuelle. Une implémentation spécifique aux besoins du JGH est donc nécessaire.

## 4 Vue d'ensemble du produit

### 4.1 Perspective du produit

La figure 2 situe EndoMine en perspective avec les autres produits mis en place dans l'environnement du JGH.

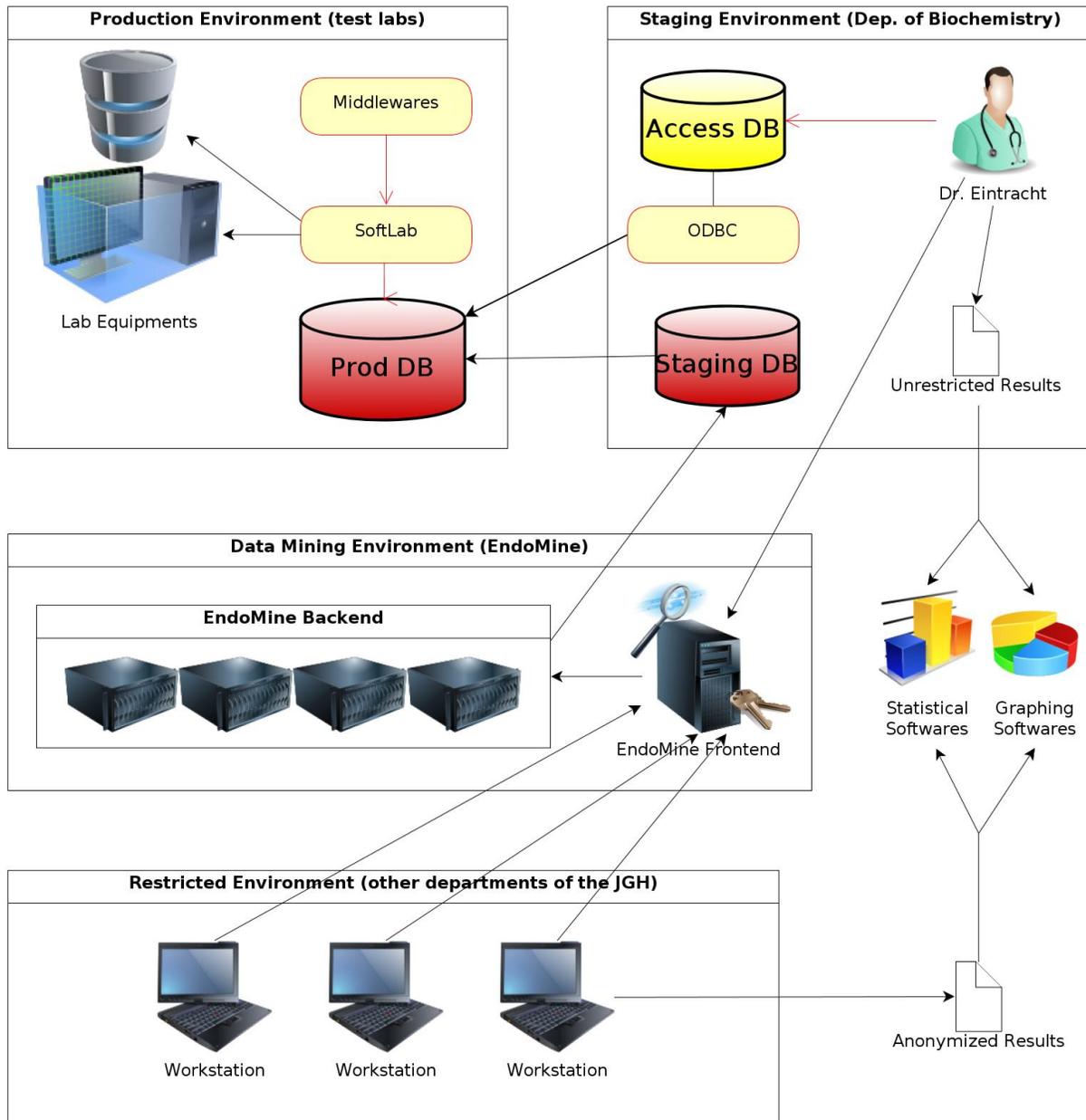


FIGURE 2 – Perspective du produit

- Le *Production Environment* représente l’environnement où les données sont collectées et la BD Oracle de production est peuplée par le logiciel SoftLab.
- La *Staging DB* est une copie miroir de la BD de production dont les informations confidentielles ont été anonymisées. **Seul le département de biochimie a accès au *Staging Environment*.**
- Les utilisateurs effectueront le forage de données sur le système EndoMine, et pourront par la suite importer leur résultats dans des logiciels de statistiques et de graphiques.

## 4.2 Sécurité et Confidentialité

Cette section décrit les différentes mesures à prendre pour protéger la confidentialité des données :

- L’accès au système EndoMine serait restreint selon les protocoles de sécurité du JGH.
- Les informations confidentielles (noms des patients, numéro de RAMQ, numéro d’hôpital, etc.) seront absentes du système EndoMine.
- EndoMine contiendra la clef anonymisée des patients, mais la correspondance entre la clef anonymisée et la fiche d’hôpital du patient ne sera pas accessible par le système EndoMine.
- Voir la section 4.4 (‘Hypothèses et dépendances’) pour plus d’informations.

## 4.3 Principaux avantages

TABLE 6: Avantages EndoMine

Bénéfices pour le client	Caractéristiques correspondantes
N1. Confidentialité et sécurité des données	HYP01, HYP03, HYP04, FEA05, FEA10, VC01, VC02, VC03
N2. Forage de données n’interférant pas avec la collection de données	HYP02, FEA05, FEA06, VC05
N3. Outil simple pour créer des requêtes	FEA07, FEA08, FEA09, FEA11, FEA12, VC04
N4. Mise à l’échelle facile du système	FEA01, FEA02, FEA03, FEA04, VC04
N5. Forage de données sans limite de mémoire	FEA02, FEA03, FEA11

## 4.4 Hypothèses et dépendances

### HYP01 BD de *staging*

La BD de *staging* sera fournie par le JGH, et sera déjà anonymisée (voir la section 4.2).

### HYP02 Synchronisation

La BD de staging sera synchronisée automatiquement une fois par jour (ou toute autre fréquence jugée acceptable par le client).

### **HYP03 Anonymisation**

Les scripts pour anonymiser / désanonymiser seront mis à la disposition du Dr. Eintracht par le JGH.

### **HYP04 Prototype**

Lors du prototype, seul le Dr. Eintracht aura accès au système EndoMine installé au JGH.

### **HYP05 Conception, développement, et test**

Les ressources de l'ÉTS doivent avoir accès aux données anonymisées et au système pilote pour les besoins de conception, développement et test.

## **4.5 Licences et installation**

1. EndoMine sera implémenté avec logiciels libres ayant prouvé leur efficacité sur des sites tels que Facebook, Amazon et Yahoo. Donc aucune licence à acheter.
2. Le code développé pour le logiciel EndoMine sera la propriété commune du JGH, de l'ÉTS et des développeurs ayant travaillé sur EndoMine. Les propriétaires du logiciel pourront installer une copie du logiciel sur d'autres serveurs et y effectuer des modifications.
3. Les données provenant de la base de données de production Oracle, ainsi que les résultats de recherches réalisés avec EndoMine sont la propriété exclusive du JGH.
4. Le système EndoMine sera installé sur le serveur du JGH par les développeurs de l'ÉTS.
5. Aucune installation requise sur les postes de travail des chercheurs.

## **5 Caractéristiques (*features*) du produit**

### **FEA01 Configuration minimale pour ajouter d'autres machines.**

L'objectif est de monter le système de telle façon que le coût de configuration / stabilité avec l'ajout d'une nouvelle machine parallèle soit minimal.

### **FEA02 Code supportant le parallélisme.**

En lien avec FEA01, le forage de données devrait pouvoir exploiter le parallélisme du système de machines distribués.

### **FEA03 Distribution automatique des données pour supporter parallélisme**

Une fois une nouvelle machine installée, le système devrait balancer automatiquement (le plus possible) la charge de traitement sur la nouvelle machine.

#### **FEA04 Supporter plusieurs requêtes en parallèles**

Plusieurs requêtes d'un même (ou plusieurs) utilisateur(s) devraient pouvoir être exécutés en parallèle.

#### **FEA05 Environnement de forage de données distinct des sources de données en production**

L'utilisation du système développe ne devrait avoir aucun impact sur le contenu ou le fonctionnement des sources de données en production.

#### **FEA06 Synchronisation des données automatique entre les sources de données et le système EndoMine**

EndoMine devrait pouvoir se synchroniser automatiquement avec les sources de données (dans l'environnement de *staging*) à intervalles régulières configurables.

#### **FEA07 Générateur de requêtes intégré**

Un générateur de requêtes permettant de faire des requêtes d'une manière interactive et simple devrait être fournis. EndoMine s'inspirera de *Microsoft Access Query Builder* (voir la Figure 3 de l'A).

#### **FEA08 Recherche par filtrage incrémentiel de la requête originale**

Modification d'une requête en ajoutant des conditions de recherche, qui ne faisait pas partie de la requête originale. C'est-à-dire de pouvoir filtrer les résultats d'une recherche avec des nouveaux critères, sans que le système aille besoin de ré-exécuter la requête originale. Effectuer un filtrage secondaire devrait prendre une fraction du temps de la requête originale. Par exemple, si on recherche les résultats de Glucose en 2010, on devrait pouvoir ajouter les résultats de Glucose en 2009 et/ou ajouter les résultats de Fructose.

#### **FEA09 Exportation des résultats de recherche**

Les résultats de recherche devraient pouvoir être exportés selon au moins 1 format d'échange de fichier d'un logiciel d'analyse statistique.

#### **FEA10 Exportation des clés anonymisées**

Permettre de facilement exporter seulement l'ensemble unique des clés anonymisées des patients présents dans les résultats d'une recherche.

#### **FEA11 Ajout de champs supplémentaire à des résultats de recherche**

À partir de résultats de recherche, il doit être possible d'ajouter des champs qui n'étaient pas inclus dans la sélection. Par exemple, on pourrait vouloir ajouter le champs sexe et âge du patient. Tous les champs disponibles dans EndoMine devraient pouvoir être ajoutés de

cette façon.

#### **FEA12 Trier les résultats de recherche**

Pouvoir trier les résultats de recherche selon n'importe quel champ inclus dans la recherche.

#### **FEA13 Dés/Anonymisation des informations confidentielles**

Pouvoir utiliser les informations anonymisées dans le système. L'utilisation devrait avoir l'option d'importer les informations anonymisées comme critère de recherche. De plus l'utilisateur devrait pouvoir exporter les informations anonymisées du système pour pouvoir les dé-anonymiser avec un autre logiciel.

#### **FEA14 Filtrage automatique des colonnes des résultats**

Permettre de filtrer les résultats de recherche. Équivalent à ajouter un filtre dans excel sur une colonne.

#### **FEA15 Extraction des valeurs uniques d'une colonne**

Pouvoir toutes les résultats uniques dans une des colonnes du résultat.

## **6 Contraintes**

### **VC01 Accessibilité**

Le système EndoMine ne doit pas être accessible à l'extérieur du réseau du JGH.

### **VC02 Confidentialité**

Les utilisateurs restreints ne peuvent accéder aux informations confidentielles sur les patients. Voir la section 4.2 (Sécurité et Confidentialité).

### **VC03 Règles du JGH**

Les règles de sécurité informatique et de confidentialité du JGH doivent être respectées.

### **VC04 Coût**

La réalisation et l'installation du prototype doit être de faible coût.

### **VC05 Modification**

La base de données de production ne doit pas être modifiée par EndoMine.

## 7 Gammes de qualité

Le forage de données doit être au minimum deux fois plus rapide que le système actuel.

## 8 Attributs des caractéristiques

Le tableau suivant permet :

- Au client de prendre connaissance des efforts et risques associés au projet.
- Aux développeurs de jauger les bénéfices et priorité que le client associe à chaque caractéristique.
- Et d'identifier les caractéristiques les plus susceptibles de changer dans le futur.

La légende des valeurs possibles pour chacune des colonnes est présenté à l'A.

TABLE 7: Attributs des caractéristiques

Caractéristiques	État	Bénéfice	Effort	Risque	Stabilité	Priorité
FEA01. Configuration minimale pour ajouter d'autres machines	Approuvé	Moyen	Moyen	Moyen	Élevé	Important
FEA02. Code supportant le parallélisme	Approuvé	Moyen	Faible	Faible	Élevé	Important
FEA03. Distribution automatique des données pour supporter parallélisme	Approuvé	Moyen	Faible	Faible	Élevé	Important
FEA04. Supporter plusieurs requêtes en parallèles	Approuvé	Élevé	Faible	Faible	Élevé	Important
FEA05. Environnement de forage de données distinct des sources de données en production	Approuvé	Élevé	Moyen	Faible	Moyen	Critique
FEA06. Synchronisation des données automatique entre les sources de données et le système EndoMine	Proposé	Moyen	Moyen	Moyen	Moyen	Important
FEA07. Générateur de requêtes intégré	Approuvé	Élevé	Élevé	Élevé	Faible	Critique
FEA08. Recherche par filtrage incrémentiel de la requête originale	Approuvé	Moyen	Élevé	Élevé	Faible	Critique
FEA09. Exportation des résultats de recherche	Proposé	Élevé	Moyen	Faible	Faible	Critique
FEA10. Exportation des clés anonymisées	Approuvé	Moyen	Faible	Faible	Moyen	Important
FEA11. Ajout de champs supplémentaire à des résultats de recherche	Approuvé	Moyen	Élevé	Élevé	Faible	Critique
FEA12. Trier les résultats de recherche	Approuvé	Faible	Moyen	Faible	Moyen	Utile
FEA13. Dés/Anonymisation des informations confidentielles	Approuvé	Élevé	Moyen	Moyen	Faible	Critique
FEA14. Filtrage automatique des colonnes des résultats	Approuvé	Élevé	Moyen	Faible	Moyen	Utile
FEA15. Extraction des valeurs uniques d'une colonne	Approuvé	Élevé	Moyen	Faible	Élevé	Utile

## 9 Autres exigences du produit

### 9.1 Exigences du système

ES1. Réquis minimaux de l'ordinateur exécutant la recherche :

- Fureteur Firefox 8 ou équivalent.
- 2 GB de mémoire vive (RAM).
- Suite de bureautique Office installée (optionnel).
- Logiciel(s) d'analyse(s) statistique(s) (optionnel).

### 9.2 Exigences de performance

L'utilisateur devrait pouvoir faire une recherche itérative en moins d'une journée avec le système.

## 10 Exigences de documentation

### 10.1 Manuel de l'utilisateur

EMU1. Un manuel d'utilisateur simple et complet, permettra à l'utilisateur d'apprendre rapidement le fonctionnement de l'interface de création de requêtes.

### 10.2 Guides d'installation, de configuration, et fichier à lire

EGCF1. Documenter les configurations requises pour installer et configurer une machine supplémentaire pour le système EndoMine.

EGCF2. Documenter les configurations requises pour ajouter une nouvelle source de données ou table supplémentaire (si le temps le permet).

## 11 Bibliographie

- [1] *Access 2010 specifications*. Microsoft. Consulté le 3 octobre 2012, à <http://office.microsoft.com/en-us/access-help/access-2010-specifications-HA010341462.aspx>
- [2] *ArchSummary Query*. 10 fév. 2010. Jessica Iannone. Consulté le 3 octobre 2012, à <http://jessica-iannone.com/img/interactive/ArchSummQry.jpg>

# Annexes

## A Microsoft Access Query Builder

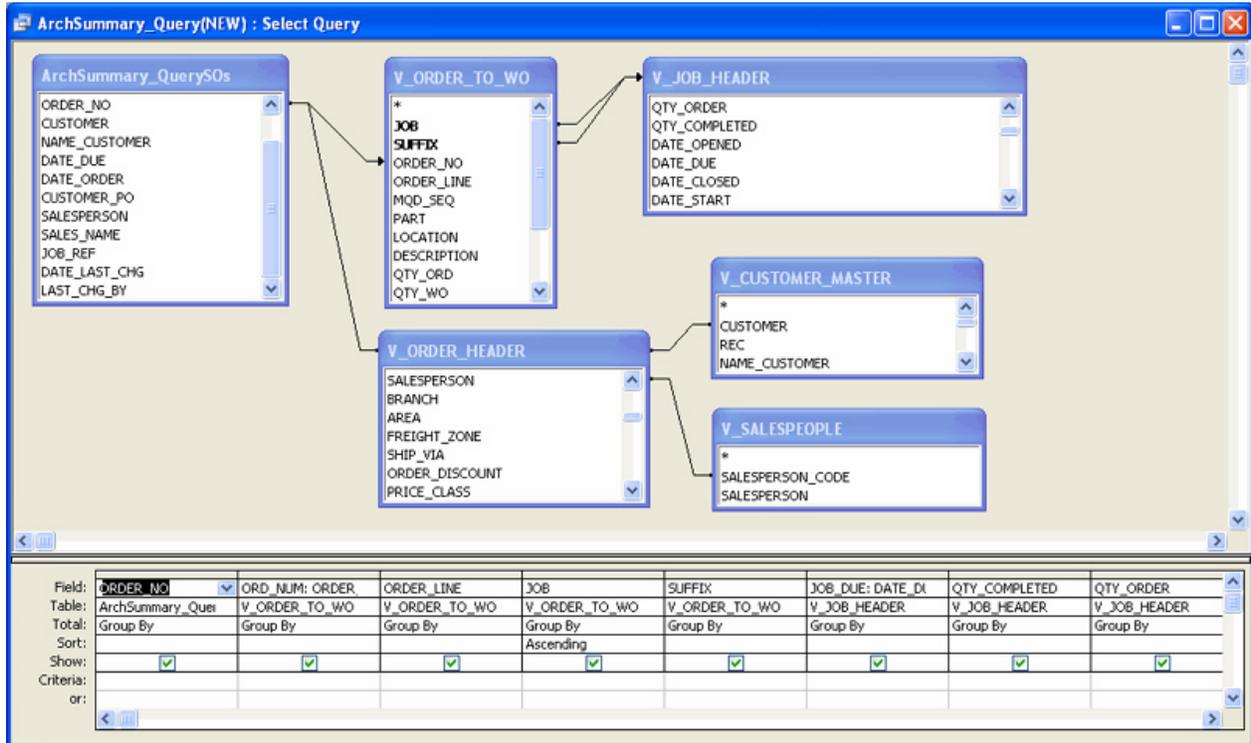


FIGURE 3 – Exemple d’une requête construite avec le Microsoft Access Query Builder (image tirée d’un siteweb : [2])

## Attributs des caractéristiques

TABLE 8 – Légende : État des caractéristiques

---

Proposé	La caractéristique est proposée, mais n'a pas encore été approuvée par les parties prenantes.
Approuvé	La caractéristique est approuvée par les parties prenantes.
Incorporé	La caractéristique est incluse dans le produit.

---

TABLE 9 – Légende : Bénéfice des caractéristiques

---

Faible	La caractéristique apporte peu de valeur ajoutée au produit et n'est pas nécessaire à son bon fonctionnement.
Moyen	La caractéristique apporte une valeur ajoutée additionnelle au produit, mais n'est pas critique à son bon fonctionnement.
Élevé	La caractéristique apporte une valeur ajoutée importante au produit et est essentielle à son bon fonctionnement ou à la réalisation de ses tâches.

---

TABLE 10 – Légende : Effort des caractéristiques

---

Faible	La réalisation de la caractéristique nécessite un effort de moins de 20 heures-personnes.
Moyen	La réalisation de la caractéristique nécessite un effort entre 20 et 40 heures-personnes.
Élevé	La réalisation de la caractéristique nécessite un effort de plus de 40 heures-personnes.

---

TABLE 11 – Légende : Risque des caractéristiques

---

Faible	La technologie utilisée et la méthode d'implémentation sont connues et bien maîtrisées.
Moyen	La technologie utilisée est récente ou la méthode d'implémentation nécessite une attention particulière.
Élevé	La technologie utilisée est nouvelle et peu éprouvée ou la méthode d'implémentation est complexe et demande une analyse plus complète.

---

TABLE 12 – Légende : Stabilité des caractéristiques

---

Faible	Les exigences concernant la caractéristique ont de fortes chances de changer ou le bon fonctionnement de la caractéristique a un impact critique sur le fonctionnement général du système et peut compromettre son exécution.
Moyen	Les exigences concernant la caractéristique sont susceptibles de changer ou le bon fonctionnement de la caractéristique a un impact sur le fonctionnement général du système sans toutefois compromettre son exécution.
Élevé	Les exigences concernant la caractéristique ont peu de chance de changer et le bon fonctionnement de la caractéristique n'a pas d'impact sur le fonctionnement général du système.

---

TABLE 14 – Légende : Priorité des caractéristiques

---

Utile	La caractéristique apporte des fonctionnalités accessoires au système. Son inclusion dans le produit a peu d'impact sur la satisfaction du client et sur l'utilisation du système.
Important	La caractéristique apporte des fonctionnalités supplémentaires au système. Son inclusion dans le produit peut influencer la satisfaction du client, mais son absence n'empêche pas l'utilisation du système.
Critique	La caractéristique est primordiale au fonctionnement du système. Il est nécessaire de l'inclure en priorité dans le produit pour assurer la totale satisfaction du client et son absence pourrait empêcher l'utilisation du système.

---

# Annexe C

## Document de SRS

# ENDOMINE

PROJET DE DÉVELOPPEMENT D'OUTILS DE FORAGE DE DONNÉES DE  
RÉSULTATS DE TESTS PATIENTS - ENDOCRINOLOGIE, MÉTABOLISME ET  
ÉPIDÉMIOLOGIE CLINIQUE

Projet #2012-076

Version : 1.0.1  
Date d'émission : 31 octobre 2012  
Date de révision : 7 novembre 2012

---

## Software Requirements Specification

---

Auteurs :



David Lauzon  
Anton Zakharov

Présenté à :



Department of  
Diagnostic Medicine

## Table des matières

<b>1</b>	<b>Introduction</b>	<b>3</b>
1.1	Objectif . . . . .	3
1.2	Portée . . . . .	3
1.3	Références . . . . .	3
1.4	Hypothèses et Dépendances . . . . .	3
<b>2</b>	<b>Survol du Modèle des Cas d'Utilisation</b>	<b>4</b>
<b>3</b>	<b>Les Acteurs</b>	<b>4</b>
3.1	Utilisateur . . . . .	4
3.2	Administrateur . . . . .	4
<b>4</b>	<b>Les Exigences</b>	<b>4</b>
4.1	Les exigences fonctionnelles . . . . .	4
	REQ001 <i>ETVX configuration ajout de machine</i> (PROJETÉ DANS LE FUTUR)	4
	REQ002 <i>Intégrer les tables principales dans le système.</i> . . . . .	4
	REQ003 <i>Widgets de liste de tables</i> . . . . .	4
	REQ004 <i>Widget pour une table</i> . . . . .	4
	REQ005 <i>Widget d'alias de table</i> (PROJETÉ DANS LE FUTUR) . . . . .	4
	REQ006 <i>Alias pour nom de colonne</i> . . . . .	4
	REQ007 <i>Widget de création de relations internes (Table A vs table B)</i> . . . . .	5
	REQ008 <i>Opérations permises sur des colonnes retournées.</i> . . . . .	5
	REQ009 <i>Opérations permises filtrage entre colonne et constante.</i> . . . . .	5
	REQ010 <i>Opérations permises de filtrage entre colonnes.</i> . . . . .	5
	REQ011 <i>Opérations de regroupement</i> . . . . .	5
	REQ012 <i>Sauvegarder une requête.</i> (PROJETÉ DANS LE FUTUR) . . . . .	5
	REQ013 <i>Autocomplete des valeurs sur des petites tables</i> (PROJETÉ DANS LE FUTUR) . . . . .	5
	REQ014 <i>(Dés)Anonymisation des informations confidentielles</i> . . . . .	5
	REQ015 <i>Exportation des résultats</i> (PROJETÉ DANS LE FUTUR) . . . . .	5
	REQ016 <i>Importation des clés anonymisées</i> (PROJETÉ DANS LE FUTUR) . . . . .	5
	REQ017 <i>Affichage des résultats</i> . . . . .	5
	REQ018 <i>Triage et filtrage des résultats.</i> (PROJETÉ DANS LE FUTUR) . . . . .	5
	REQ019 <i>Multi-utilisateurs / Multi-requêtes</i> (PROJETÉ DANS LE FUTUR) . . . . .	5
	REQ020 <i>Effacer des requêtes sauvegardés</i> . . . . .	5
4.2	Les Exigences Non Fonctionnelles . . . . .	6
4.2.1	Functionality > Accuracy . . . . .	6
	NFR001 L'utilisateur utilise l'application et 1 des disques dur du serveur 'plante'. . . . .	6
4.2.2	Usability > Operability . . . . .	6
	NFR002 L'utilisateur compose une requête type (présentée dans le SRS comme exemple). . . . .	6
	NFR003 L'utilisateur procède à une requête qui demande un scan complet des tables. . . . .	6
4.2.3	Maintainability > Changeability . . . . .	6
	NFR004 L'administrateur veut ajouter une autre machine au cluster. . . . .	6

NFR005	Le programmeur veut ajouter une autre petite table dans le système . . .	6
<b>5</b>	<b>Documentation en direct pour l'utilisateur et exigences du système d'aide</b>	<b>6</b>
<b>6</b>	<b>Contraintes de conception</b>	<b>6</b>
CR001	<i>Distribution automatique des données pour supporter parallélisme . . . . .</i>	7
CR002	<i>Code supportant le parallélisme . . . . .</i>	7
CR003	<i>Synchronisation automatique des données entre les sources de données et le système EndoMine (PROJETÉ DANS LE FUTUR) . . . . .</i>	7
CR004	<i>Générateur de recherche doit fonctionner sur plusieurs systèmes operationnels . . . . .</i>	7
CR005	<i>ETL : Grouper les ids référant aux même patients . . . . .</i>	7
CR006	<i>ETL : Uniformiser les valeurs des données de la table v_p_test_result . . . . .</i>	7
CR007	<i>Trier les données . . . . .</i>	7
CR008	<i>Limite de taille de résultats affichés . . . . .</i>	7
CR009	<i>Changement des schéma, synchronisation des données dans EndoMine. (PROJETÉ DANS LE FUTUR) . . . . .</i>	7
CR010	<i>Ajout/suppression des nouvelles tables (PROJETÉ DANS LE FUTUR) . . . . .</i>	7
CR011	<i>Limite de taille de jointure des tables. . . . .</i>	7
CR012	<i>Limite des résultats retournés. . . . .</i>	7
CR013	<i>License et maintenance du logiciel. . . . .</i>	8
<b>7</b>	<b>Interfaces</b>	<b>9</b>
7.1	Interfaces Utilisateur . . . . .	9
7.2	Interfaces Logicielles . . . . .	9
7.3	Interfaces de Communications . . . . .	9
<b>8</b>	<b>Remarques légales, de droits d'auteur, et diverses</b>	<b>11</b>
<b>9</b>	<b>Normes et Standards Applicables</b>	<b>11</b>
9.1	Quint-2 . . . . .	11
<b>10</b>	<b>Bibliographie</b>	<b>12</b>
<b>A</b>	<b>Spécifications des Cas d'Utilisation</b>	<b>13</b>
A.1	UC01 : Query A . . . . .	13
A.2	UC02 : Query B . . . . .	13
A.3	UC03 : Panic Results (getting time of panic results) . . . . .	14
A.4	UC04 : See all people coming to each clinic . . . . .	14
<b>B</b>	<b>Matrice de Tracabilité</b>	<b>16</b>
<b>C</b>	<b>Diagrammes UML</b>	<b>18</b>
<b>D</b>	<b>Légendes des diagrammes UML</b>	<b>20</b>
<b>E</b>	<b>License BSD</b>	<b>22</b>

## Liste des tableaux

1	Historique des Révisions . . . . .	2
2	Matrice de Tracabilité . . . . .	16
2	Matrice de Tracabilité . . . . .	17

## Table des figures

1	Création d'une requête Hadoop . . . . .	9
2	Naviguer dans l'arborescence de fichiers Hadoop (Hue) . . . . .	10
3	Prévisualisation du contenu d'un fichier Hadoop . . . . .	10
4	Télécharger ou effacer des résultats de recherche . . . . .	11
5	Diagramme d'Activité AD01 - Importation des données dans Hadoop . . . . .	19
6	Légende des diagrammes de cas d'utilisation (référence [2]) . . . . .	20
7	Légende des diagrammes de séquence (référence [2]) . . . . .	20
8	Légende des diagrammes d'activité (référence [2]) . . . . .	21
9	Légende des modèles du domaine (référence [2]) . . . . .	21
10	Légende des diagrammes d'activité . . . . .	22

# Historiques des Révisions

TABLE 1: Historique des Révisions

Date	Version	Description	Auteur
7/nov/2012	v1.0.1	Changement du departement et correction sur section de licence	David Lauzon
5/nov/2012	v1.0	Ajout des contraintes et licence	Anton Zakharov
31/oct/2012	v0.8.1-rc1	Polissage du rapport. Correction des erreurs de tout genre.	Anton Zakharov,David Lauzon
31/oct/2012	v0.8	Correction de la matrice de traca- bilité, ajout de la section 12	Anton Zakharov
31/oct/2012	V0.7	Ajout des sections 1, 5, Annexe A	David Lauzon
30/oct/2012	V0.6	Correction des réquis	Anton Zakharov
30/oct/2012	v0.5	Section 7, 9, 10, 11	Anton Zakharov
29/oct/2012	v0.4	Section 6	David Lauzon
29/oct/2012	v0.3	Section 4	Anton Zakharov,David Lauzon
29/oct/2012	v0.2	Sections 2 et 3	Anton Zakharov
29/oct/2012	v0.1	Gabarit initial	David Lauzon

## 1 Introduction

### 1.1 Objectif

Le but de ce document est de ressembler d'analyser, et de définir les exigences du système de forage de données EndoMine. EndoMine sera un outil de forage de données pour le département de biochimie médicale de l'hôpital général juif (JGH). Ce document détermine les exigences fonctionnelles et non fonctionnelles nécessaires pour réaliser le produit spécifié par le client dans le document de Vision.

### 1.2 Portée

La portée de SRS visera à répondre aux besoins du Dr. Shaun Eintracht (STK2). Les besoins des autres intervenants seront répondus dans un SRS subséquent.

Pour le positionnement du système, voir la Figure "Perspective du Produit" dans le document de Vision.

### 1.3 Références

Le document de vision : EndoMine\_-\_Vision.pdf

### 1.4 Hypothèses et Dépendances

Lors du prototype, seul le Dr. Eintracht aura accès au système EndoMine installé au JGH.

## 2 Survol du Modèle des Cas d'Utilisation

Le Dr. Shaun Eintracht a fourni 4 requêtes qui serviront de cas d'utilisation. Les cas complets se trouvent dans l'annexe A.

## 3 Les Acteurs

### 3.1 Utilisateur

Personne utilisant le système pour faire du forage de données (dans le cas de ce prototype, le Dr. Eintracht en est l'unique utilisateur).

### 3.2 Administrateur

Personne s'assurant du bon fonctionnement des diverses parties opérationnelles du système d'EndoMine.

## 4 Les Exigences

### 4.1 Les exigences fonctionnelles

Voici une liste des exigences, en termes de fonctionnalités, du logiciel. Ceux-ci décrivent des fonctionnalités non-incluses dans les cas d'utilisation.

#### **REQ001** *ETVX configuration ajout de machine* (PROJETÉ DANS LE FUTUR)

Une procédure devrait être fournie pour ajouter une nouvelle machine au «cluster» sur lequel le système est exécuté. Le système devrait pouvoir être distribué sur cette machine.

#### **REQ002** *Intégrer les tables principales dans le système.*

Le système devrait contenir les données des tables suivantes du système original: v\_p\_lab\_stay; v\_p\_lab\_patient; v\_p\_lab\_order; v\_p\_lab\_specimen; v\_s\_lab\_specimen; v\_s\_lab\_test; v\_s\_lab\_clinic; v\_s\_lab\_doctor. Les exceptions sont les données qui sont relatives aux données des patients.

#### **REQ003** *Widgets de liste de tables*

Le système offre un widget énumérant toutes les tables. L'utilisateur peut sélectionner une ou plusieurs tables à partir de ce widget.

#### **REQ004** *Widget pour une table*

Le système offre un widget représentant une table. L'utilisateur peut sélectionner des colonnes qui seront affichées / travaillées avec.

#### **REQ005** *Widget d'alias de table* (PROJETÉ DANS LE FUTUR)

Le système permet de sélectionner la même table plusieurs fois et la renommer.

#### **REQ006** *Alias pour nom de colonne*

Le système doit permettre à l'utilisateur de faire un alias sur le nom d'une colonne en output.

**REQ007** *Widget de création de relations internes (Table A vs table B)*

Le système offre un widget permettant de faire des relations de jointure interne entre une paire de tables.

**REQ008** *Opérations permises sur des colonnes retournées.*

Les colonnes retournées devraient permettre des opérations simples de formatage. Ceux-ci incluent la concaténation; convertissement en types primaires ; manipulation de date et des fonctions comme SUM ou COUNT. De plus l'utilisateur devrait pouvoir choisir quelles colonnes seront retournées dans le resultat.

**REQ009** *Opérations permises filtrage entre colonne et constante.*

Les opérations suivantes devraient être permises pour filtrer entre colonnes et constantes :<;>;=;LIKE; NOT LIKE; AND;OR;! =; BETWEEN; IS NULL; IS NOT NULL; DISTINCT

**REQ010** *Opérations permises de filtrage entre colonnes.*

les opérations de filtrage entre colonnes suivantes sont permises : <;>;=;! =

**REQ011** *Opérations de regroupement*

Le système devrait permettre de regrouper les données pour une ou plusieurs colonnes.

**REQ012** *Sauvegarder une requête. (PROJETÉ DANS LE FUTUR)*

Le système devrait permettre de sauvegarder une sous-requête (SQL et/ou DONNÉES) afin de la réexécuter/visualiser plus tard.

**REQ013** *Autocomplete des valeurs sur des petites tables (PROJETÉ DANS LE FUTUR)*

Faire un autocomplete pour les valeurs des colonnes des petites tables ( < 2Mb en taille)

**REQ014** *(Dés)Anonymisation des informations confidentielles*

Les id des patients entrées dans le système doivent être anonymisées. Seulement le sexe et la date de naissance du patient devraient se retrouver dans le système. Les données anonymisées ne devraient pas pouvoir être dé-anonymisés par le système .

**REQ015** *Exportation des résultats (PROJETÉ DANS LE FUTUR)*

Les résultats de recherche devraient pouvoir être téléchargés par l'utilisateur.

**REQ016** *Importation des clés anonymisées (PROJETÉ DANS LE FUTUR)*

Le système devrait pouvoir importer les clés anonymisées (dans le cas qu'il y a beaucoup de clés) pour pouvoir les utiliser pour filtrer la recherche.

**REQ017** *Affichage des résultats*

Le système devrait afficher les résultats de recherche.

**REQ018** *Triage et filtrage des résultats. (PROJETÉ DANS LE FUTUR)*

Le système devrait permettre de trier et filtrer les résultats affichés.

**REQ019** *Multi-utilisateurs / Multi-requêtes (PROJETÉ DANS LE FUTUR)*

Le système devrait supporter plusieurs utilisateurs / requetes simultanément.

**REQ020** *Effacer des requêtes sauvegardés*

Les requêtes sauvegardées devraient pouvoir être effacées.

## 4.2 Les Exigences Non Fonctionnelles

Les exigences qui ne sont pas liées aux fonctionnalités du logiciel, mais plus à sa qualité. Ces exigences sont regroupées en fonction du modèle de qualité Quint-2 (expliquée à la section 9.1).

### 4.2.1 Functionality > Accuracy

**NFR001 L'utilisateur utilise l'application et 1 des disques dur du serveur 'plante'.**  
(1.2.1. Failure ratio)

Le système est en opération journalière. Le système continue de traiter la requête du client d'une façon transparente 100% du temps..

### 4.2.2 Usability > Operability

**NFR002 L'utilisateur compose une requête type (présentée dans le SRS comme exemple).**  
(3.3.21. Ultimate operation time)

Le système est en opération journalière. L'utilisateur termine la création de la requête en moins de 10 min avec l'interface utilisateur.

**NFR003 L'utilisateur procède à une requête qui demande un scan complet des tables.**  
(3.3.26. Response time for user)

Le système est en opération journalière. Le système complète le traitement de la requête, et renvoie les résultats en moins de 30 min (avec 1 machine du cluster).

### 4.2.3 Maintainability > Changeability

**NFR004 L'administrateur veut ajouter une autre machine au cluster.** (5.2.1. Modification effort per unit volume)

Le système est en maintenance. La machine ajoutée au cluster et est opérationnelle avec le système EndoMine en moins de 4 heures.

**NFR005 Le programmeur veut ajouter une autre petite table dans le système**  
(5.2.1. Modification effort per unit volume)

Le système est en maintenance. La table est importée et accessible dans le système EndoMine en moins de 2 heures.

## 5 Documentation en direct pour l'utilisateur et exigences du système d'aide

Pas de documentation dans le logiciel en tant que tel. Mais un manuel d'utilisateur sera disponible pour supporter l'utilisation du logiciel (plus de détails dans le document de vision).

## 6 Contraintes de conception

Voici la liste de contraintes que le système doit respecter :

**CR001** *Distribution automatique des données pour supporter parallélisme*

Le système devrait automatiquement distribuer des données à la nouvelle machine (distribuer les données à travers le réseau)

**CR002** *Code supportant le parallélisme*

Le logiciel utilisé pour faire les requêtes doit être optimisé pour travailler sur plusieurs machines.

**CR003** *Synchronisation automatique des données entre les sources de données et le système EndoMine (PROJETÉ DANS LE FUTUR)*

Les données devraient être synchronisé automatiquement entre l'environnement de staging et l'environnement de data mining.

**CR004** *Générateur de recherche doit fonctionner sur plusieurs systèmes operationnels*

Le générateur de recherche doit fonctionner sur Windows et Linux ayant un navigateur récent (année 2010+)

**CR005** *ETL: Grouper les ids référant aux même patients*

Étant donné qu'un même patient peut avoir plusieurs PatientID, il faudrait le nettoyer et grouper les patients qui sont la même personne. Notre solution temporaire serait d'ajouter un champ dans qui permettra d'indiquer quels sont les Ids en double référant à la même personne. Nous ne sommes pas responsables de le remplir.

**CR006** *ETL: Uniformiser les valeurs des données de la table v\_p\_test\_result*

Les valeurs des champs dans v\_p\_test\_result n'ont pas la même signification due aux différentes méthodes utilisés pour les cueillir. Nous allons ajouter un champ dans lequel il serait possible d'indiquer si les valeurs sont "clean" où non. Nous ne somme pas responsable de remplir ce champ.

**CR007** *Trier les données*

La version du générateur de requêtes fournie ne permettrait pas de trier les données dans une requête.

**CR008** *Limite de taille de résultats affichés*

L'interface graphique affichera à l'écran les 4000 premiers octets des résultats.

**CR009** *Changement des schéma, synchronisation des données dans EndoMine. (PROJETÉ DANS LE FUTUR)*

Les changements de structure aux tables ainsi que la mise à jour ou suppression des données existantes (update / delete) n'est pas supportée. Dans l'itération 1, il faudra effacer la BD d'EndoMine et refaire manuellement une importation complète de toutes les données.

**CR010** *Ajout/suppression des nouvelles tables (PROJETÉ DANS LE FUTUR)*

L'ajout / suppression des nouvelle tables dans le système. (Devrait être fait manuellement dans la première itération)

**CR011** *Limite de taille de jointure des tables.*

La plus grosse table doit être spécifiée en premier. Toutes les autres tables à joindre doivent pouvoir être lue au complet dans la mémoire du serveur d'EndoMine.

**CR012** *Limite des résultats retournés.*

EndoMine supporte l'exportation de résultats jusqu'à 100MB en taille.

**CR013** *License et maintenance du logiciel.*

Le logiciel fourni lors de l'itération 1 (30 novembre 2012) est une preuve de concept. Comme avec toutes les versions prototypes, il peut exister des bogues, mais nous assurons que nous allons tester au meilleur de nos capacités dans les délais disponibles. Consulter l'ÉTS pour les itérations futures. Le logiciel est fourni sous la license BSD (voir annexe).

## 7 Interfaces

### 7.1 Interfaces Utilisateur

L'utilisateur accède au système EndoMine par le biais d'une interface web. Une fois qu'il est authentifié il peut commencer à créer des requêtes (Fig 1).

Ensuite à l'aide de l'utilitaire Hue, il peut voir la liste des recherches complétées (Fig 2), ainsi que prévisualiser les premières lignes de leur contenu (Fig 3). Aussi, il peut également télécharger ou supprimer des résultats de recherches (Fig 4).

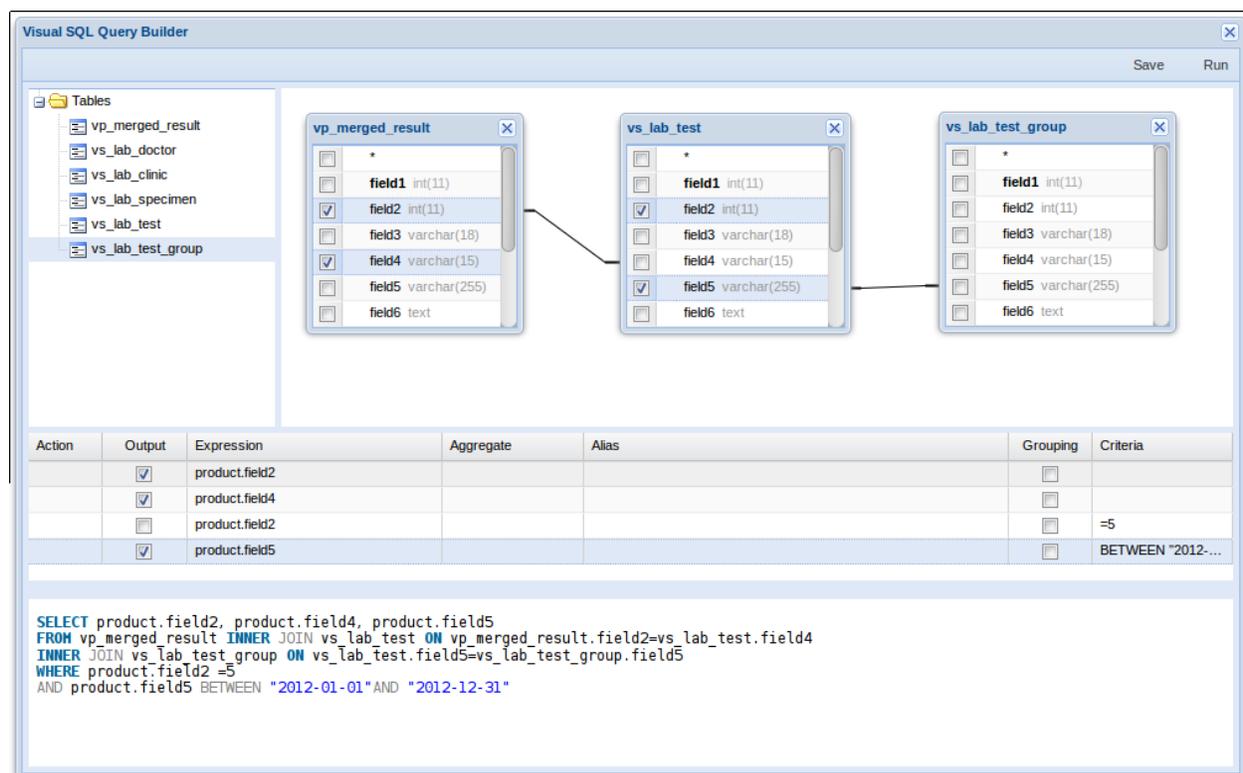


FIGURE 1 – Création d'une requête Hadoop

### 7.2 Interfaces Logicielles

Cette section est définie plus en détails dans le document d'architecture.

### 7.3 Interfaces de Communications

Cette section est définie plus en détails dans le document d'architecture.

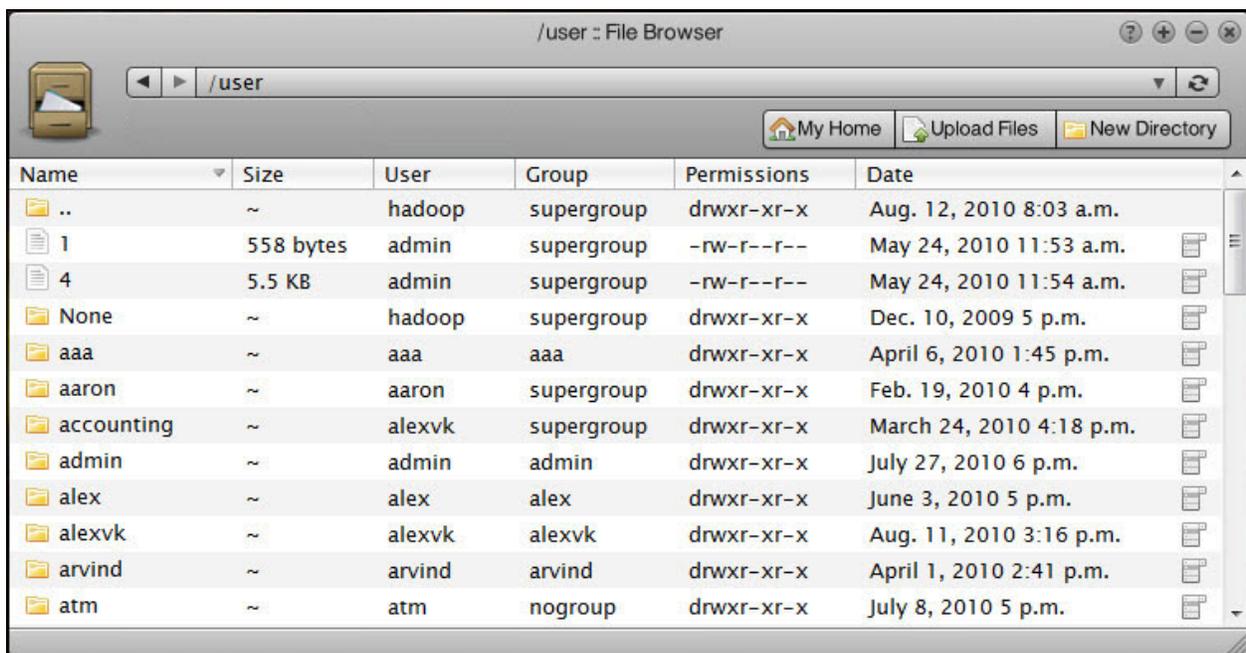


FIGURE 2 – Naviguer dans l'arborescence de fichiers Hadoop (Hue)

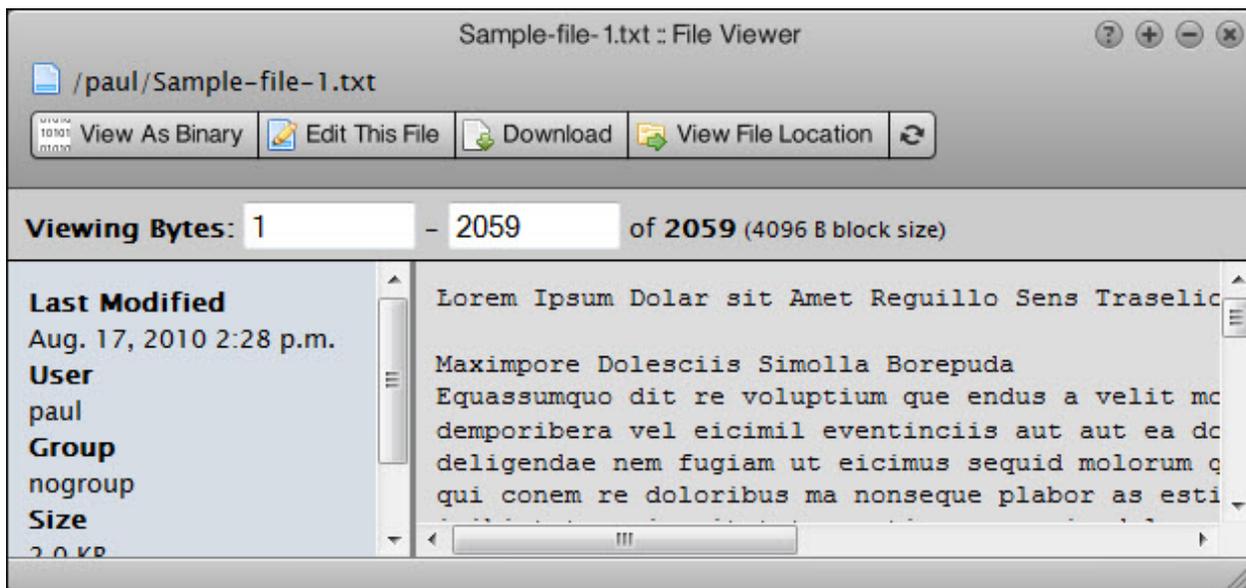


FIGURE 3 – Prévisualisation du contenu d'un fichier Hadoop

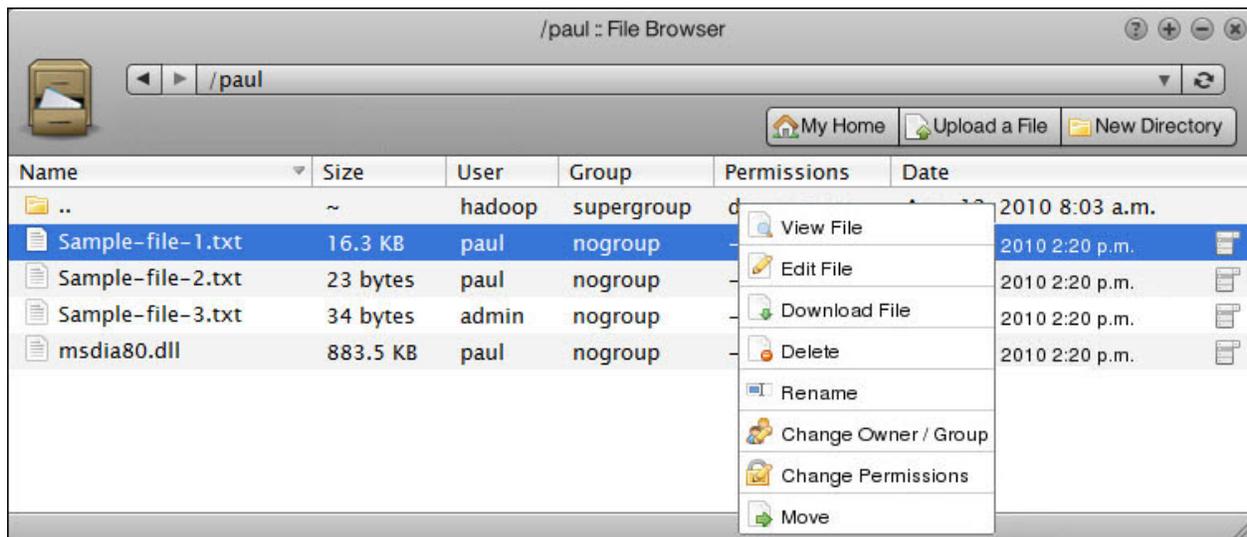


FIGURE 4 – Télécharger ou effacer des résultats de recherche

## 8 Remarques légales, de droits d’auteur, et diverses

Le code d’EndoMine, ainsi que tous les documents livrés par l’ÉTS au cours de ce projet sont couverts par la licence BSD (voir Annexe E). Les données provenant de la base de données de production Oracle, ainsi que les résultats de recherches réalisés avec EndoMine sont la propriété exclusive du département de médecine diagnostique du JGH.

## 9 Normes et Standards Applicables

### 9.1 Quint-2

Les exigences non-fonctionnelles (section 4.2) ont été classifiées selon le modèle de qualité du standard Quint-2 (Extended ISO 9126-1 Model of Software Quality). Ce modèle constituait le modèle le plus complet lors de la rédaction de ce document, car il classifie les exigences en cinq grandes classes : Functionality, Reliability, Efficiency, Maintainability, et Portability.

Pour plus d’information, se référer à : <http://www.serc.nl/quint-book/>

## 10 Bibliographie

- [1] Cloudera. (2012). *Introducing Hue File Browser*. Consulté le 31 octobre 2012, à <https://ccp.cloudera.com/display/CDHDOC/File+Browser#FileBrowser-IntroducingHueFileBrowser>
- [2] LOG410 - Analyse de besoins et spécifications. (26 mars 2009). *Underwater Systems Corps - SRS*. Consulté le 7 novembre 2010, à <https://cours.etsmtl.ca/log410/private/Labo%203/Exemple-SRS-Editeur-etats.pdf>

# Annexes

## A Spécifications des Cas d'Utilisation

### A.1 UC01 : Query A

Listing 1 – UC01 : Query A

```
1 SELECT
2     DISTINCT vp_lab_order.ID ,
3     vp_lab_patient.ID ,
4     vp_lab_patient.SEX ,
5     vp_lab_stay.CLINIC_ID ,
6     vp_lab_stay.DOCTOR_ID ,
7     vp_lab_test_result.TEST_ID ,
8     vp_lab_test_result.RESULT ,
9     vp_lab_test_result.TEST_DATE ,
10    vp_lab_test_result.TEST_TIME
11 FROM
12 (
13     (
14         vp_lab_test_result
15         INNER JOIN vp_lab_order
16         ON vp_lab_test_result.ORDER_AA_ID = vp_lab_order.AA_ID
17     )
18     INNER JOIN
19     (
20         vp_lab_stay
21         INNER JOIN vp_lab_patient
22         ON vp_lab_stay.PATIENT_AA_ID = vp_lab_patient.AA_ID
23     )
24     ON vp_lab_order.STAY_AA_ID = vp_lab_stay.AA_ID
25 )
26 INNER JOIN vp_lab_specimen
27 ON (vp_lab_test_result.SPECIMEN_TYPE = vp_lab_specimen.SPECIMEN_TYPE)
28     AND (vp_lab_order.AA_ID = vp_lab_specimen.ORDER_AA_ID)
29 WHERE (((vp_lab_order.ID) Between "F0010000" And "H6319999"));
```

### A.2 UC02 : Query B

Listing 2 – UC0B : Query B

```
1 SELECT DISTINCT vp_lab_order.id ,
2     vp_lab_patient.id ,
3     vp_lab_patient.first_name ,
4     vp_lab_patient.last_name ,
5     vp_lab_patient.sex ,
6     vp_lab_stay.vs_lab_clinic_id ,
7     vp_lab_stay.doctor_id ,
8     vp_lab_test_result.ordering_workstation_id ,
9     vp_lab_test_result.testing_workstation_id ,
10    vp_lab_test_result.group_test_id ,
11    vp_lab_test_result.test_id ,
12    vp_lab_test_result.result ,
13    vp_lab_order.priority ,
14    vs_lab_clinic.vp_lab_patient_type
15 FROM vs_lab_clinic
16     INNER JOIN ((vp_lab_test_result
```

```
17         INNER JOIN vp_lab_order
18             ON vp_lab_test_result.order_aa_id = vp_lab_order.aa_id)
19     INNER JOIN (vp_lab_stay
20         INNER JOIN vp_lab_patient
21             ON vp_lab_stay.vp_lab_patient_aa_id = vp_lab_patient
22                 .aa_id)
23         ON vp_lab_order.vp_lab_stay_aa_id = vp_lab_stay.aa_id)
24     INNER JOIN vp_lab_specimen
25         ON ( vp_lab_order.aa_id = vp_lab_specimen.order_aa_id )
26         AND ( vp_lab_test_result.vp_lab_specimen_type =
27             vp_lab_specimen.vp_lab_specimen_type )
28     ON vs_lab_clinic.vs_lab_clinic_code = vp_lab_stay.vs_lab_clinic_id
29 WHERE ( ( ( vp_lab_order.id ) BETWEEN "g0010000" AND "h2319999" )
30     AND ( ( vp_lab_test_result.test_id ) = "homo"
31         OR ( vp_lab_test_result.test_id ) = "b12i" ) ) );
```

### A.3 UC03 : Panic Results (getting time of panic results)

Listing 3 – UC03 : Panic Results (getting time of panic results)

```
1 SELECT DISTINCT vp_lab_order.id ,
2                 vp_lab_test_result.test_id ,
3                 vp_lab_test_result.result ,
4                 vp_lab_test_result.test_time ,
5                 vp_lab_order.ordered_time ,
6                 vp_lab_test_result.verified_time ,
7                 INT(Val([receive_time]) / 100) * 60 + Val(RIGHT([receive_time], 2)) AS RT,
8                 INT(Val([verified_time]) / 100) * 60 + Val(RIGHT([verified_time], 2)) AS VT,
9                 INT(Val([ordered_time]) / 100) * 60 + Val(RIGHT([ordered_time], 2)) AS OT,
10                INT(Val([test_time]) / 100) * 60 + Val(RIGHT([test_time], 2)) AS TT,
11                [vt] - [rt] AS MPA_TEST,
12                [vt] - [ot] AS INLAB,
13                [vt] - [tt] AS CALL_TIME,
14                vp_lab_patient.id ,
15                vp_lab_stay.vs_lab_clinic_id ,
16                vp_lab_order.priority ,
17                vs_lab_clinic.vp_lab_patient_type ,
18                vp_lab_test_result.panic_low ,
19                vp_lab_test_result.panic_high
20 FROM vs_lab_clinic
21     INNER JOIN ((vp_lab_test_result
22         INNER JOIN vp_lab_order
23             ON vp_lab_test_result.order_aa_id = vp_lab_order.aa_id)
24         INNER JOIN (vp_lab_stay
25             INNER JOIN vp_lab_patient
26                 ON vp_lab_stay.vp_lab_patient_aa_id = vp_lab_patient
27                     .aa_id)
28             ON vp_lab_order.vp_lab_stay_aa_id = vp_lab_stay.aa_id)
29         INNER JOIN vp_lab_specimen
30             ON ( vp_lab_order.aa_id = vp_lab_specimen.order_aa_id )
31             AND ( vp_lab_test_result.vp_lab_specimen_type =
32                 vp_lab_specimen.vp_lab_specimen_type )
33         ON vs_lab_clinic.vs_lab_clinic_code = vp_lab_stay.vs_lab_clinic_id
34 WHERE ( ( ( vp_lab_order.id ) BETWEEN "h4010000" AND "h4319999" )
35     AND ( ( vs_lab_clinic.vp_lab_patient_type ) = "h"
36         OR ( vs_lab_clinic.vp_lab_patient_type ) = "u" ) ) );
```

### A.4 UC04 : See all people coming to each clinic

Listing 4 – UC04 : See all people coming to each clinic

```
1 SELECT vp_lab_order.id ,
2         vp_lab_patient.social_security ,
3         vp_lab_stay.vs_lab_clinic_id ,
4         vp_lab_stay.doctor_id
5 FROM (vp_lab_test_result
6        INNER JOIN vp_lab_order
7              ON vp_lab_test_result.order_aa_id = vp_lab_order.aa_id)
8 INNER JOIN (vp_lab_stay
9            INNER JOIN vp_lab_patient
10              ON vp_lab_stay.vp_lab_patient_aa_id = vp_lab_patient.aa_id)
11        ON vp_lab_order.vp_lab_stay_aa_id = vp_lab_stay.aa_id
12 GROUP BY vp_lab_order.id ,
13          vp_lab_patient.social_security ,
14          vp_lab_stay.vs_lab_clinic_id ,
15          vp_lab_stay.doctor_id
16 HAVING ( ( vp_lab_order.id ) BETWEEN "g0010000" AND "h1319999" )
17        AND ( ( vp_lab_stay.vs_lab_clinic_id ) LIKE "rc*" ) );
```

## B Matrice de Tracabilité

TABLE 2: Matrice de Tracabilité

ID	Caractéristique	Exigence	Exigences non fonctionnelles	Cas d'utilisations
FEA01	Configuration minimale pour ajouter d'autres machines	REQ001	NFR004	
FEA02	Code supportant le parallélisme		NFR003	
FEA03	Distribution automatique des données pour supporter le parallélisme		NFR001, NFR004	
FEA04	Supporter plusieurs requêtes en parallèles	REQ019		
FEA05	Environnement de forage de données distinct des sources de données en production	REQ002		
FEA06	Synchronisation des données automatique entre les sources de données et le système EndoMine		NFR005	
FEA07	Générateur de requêtes intégré	REQ003, REQ004, REQ005, REQ006, REQ007, REQ008, REQ009, REQ010, REQ011		UC001, UC002, UC003, UC004
FEA08	Recherche par filtrage incrémentiel de la requête originale	REQ012, REQ020		
FEA09	Exportation des résultats de recherche	REQ015		
FEA10	Exportation des clés anonymisées	REQ014		
FEA11	Ajout de champs supplémentaire à des résultats de recherche	REQ012		

TABLE 2: Matrice de Tracabilité

ID	Caractéristique	Exigence	Exigences non fonctionnelles	Cas d'utilisations
FEA12	Trier les résultats de recherche	REQ018		
FEA13	Dés/Anonymisation des informations confidentielles	REQ014		
FEA14	Filtrage automatique des colonnes des résultats	REQ018		
FEA15	Extraction des valeurs uniques d'une colonne	REQ018		

---

## C Diagrammes UML

Voir les légendes suivantes :

- Légende des diagrammes d'activité (figure 10)

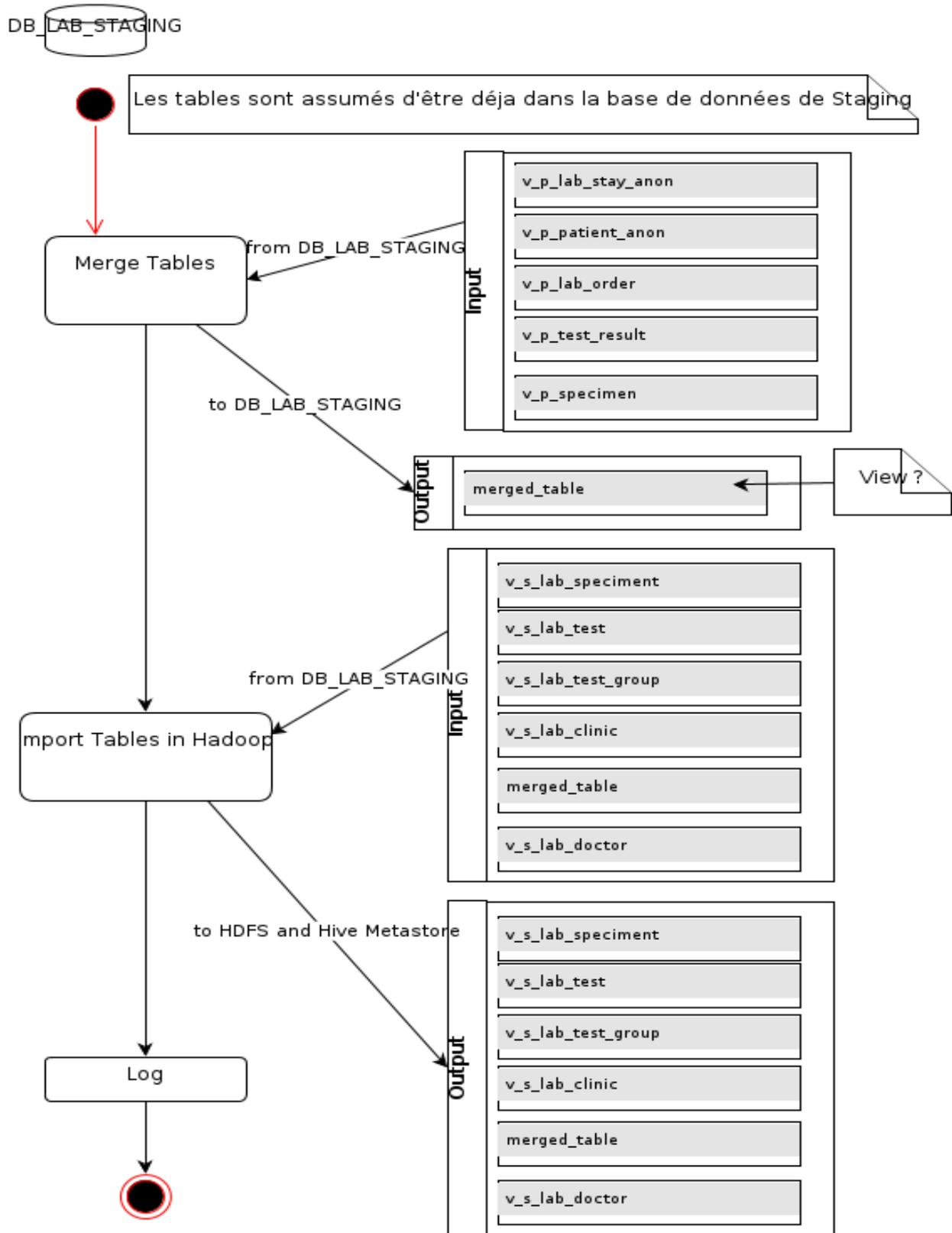


FIGURE 5 – Diagramme d'Activité AD01 - Importation des données dans Hadoop  
 Jewish General Hospital EndoMine\_-\_SRS.pdf Page 21/ 22

## D Légendes des diagrammes UML

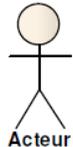
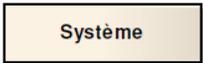
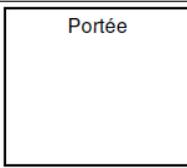
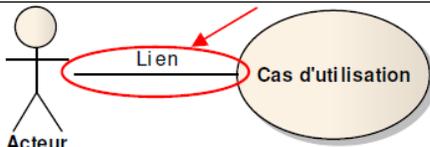
Légende	
 Acteur	Acteur humain du système
 Système	Acteur système (sous-système) du système
 Portée	Ce qui est inclus dans le système, c'est-à-dire, la limite du système et/ou la portée du système.
 Cas d'utilisation	Cas d'utilisation qui permet à un acteur d'effectuer une tâche à l'aide du système.
 Acteur	Signifie que l'acteur utilise le cas d'utilisation ou que le cas d'utilisation utilise le système externe, si le lien est pointillé.

FIGURE 6 – Légende des diagrammes de cas d'utilisation (référence [2])

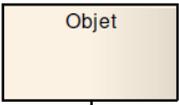
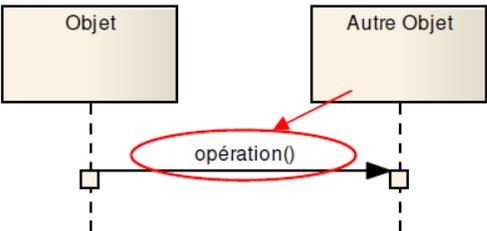
Légende	
 Objet	Objet du système qui exécute des opérations.
 Objet	L'opération exécutée sur l' « Autre objet ».

FIGURE 7 – Légende des diagrammes de séquence (référence [2])

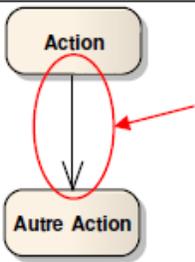
Légende	
	Début et fin du diagramme.
	Action effectuée à un moment précis.
	Décision que le système doit prendre.
	Flux de contrôle du système.

FIGURE 8 – Légende des diagrammes d’activité (référence [2])

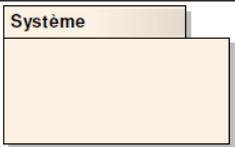
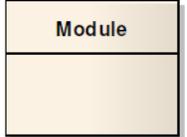
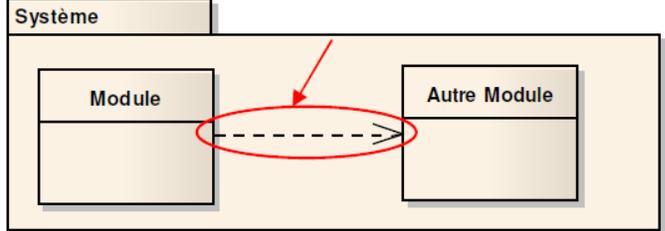
Légende	
	Système ou sous-système faisant parti du projet
	Module faisant partie d’un système ou sous-système.
	Lien entre les systèmes, sous-systèmes ou modules.

FIGURE 9 – Légende des modèles du domaine (référence [2])

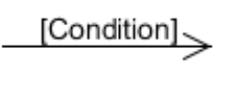
Légende	
	Début et fin de du diagramme
	L'état d'un objet à un moment précis
	Décision quand il y a plusieurs changements d'état possible pour le même flux de changement
	Représente le flux de changement d'états du système La condition représente la condition de changement d'état

FIGURE 10 – Légende des diagrammes d'activité

## E License BSD

### Listing 5 – License BSD

```
1 Copyright (c) 2012, Anton Zakharov, David Lauzon, ETS
2 All rights reserved.
3
4 Redistribution and use in source and binary forms, with or without
5 modification, are permitted provided that the following conditions are met:
6
7 1. Redistributions of source code must retain the above copyright notice, this
8   list of conditions and the following disclaimer.
9 2. Redistributions in binary form must reproduce the above copyright notice,
10  this list of conditions and the following disclaimer in the documentation
11  and/or other materials provided with the distribution.
12
13 THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
14 ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
15 WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
16 DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR
17 ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
18 (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
19 LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
20 ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
21 (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
22 SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
23
24 The views and conclusions contained in the software and documentation are those
25 of the authors and should not be interpreted as representing official policies,
26 either expressed or implied, of the FreeBSD Project.
```

# Annexe D

## Document d'architecture (brouillon)

# **Architecture EndoMine**

Hôpital Général Juif

version 0.2.1 (DRAFT)

**Anton Zakharov  
David Lauzon**

**École de technologie supérieure  
le 31 octobre 2012**

# Tables des matières

## [1 Introduction](#)

### [1.1 Version "DRAFT" du document](#)

## [2 Architecture d'EndoMine](#)

### [2.1 Composantes d'EndoMine](#)

#### [2.1.1 Diagrammes des composantes](#)

#### [2.1.2 Processus d'anonymisation](#)

### [2.2 Déploiement d'EndoMine](#)

#### [2.2.1 Diagrammes de déploiement](#)

## [3 Architecture Hadoop](#)

### [3.1 Hive](#)

### [3.2 HDFS](#)

### [3.3 Impala](#)

## [Bibliographie](#)

# 1 Introduction

Ce document décrit de façon sommaire les décisions architecturales importantes pour le projet.

## 1.1 Version “*DRAFT*” du document

Ce document est en cours d’élaboration (*DRAFT*) pour permettre aux administrateurs système du JGH d’avoir une idée des implications nécessaires à mettre en place au niveau de la sécurité informatique. Une version plus exhaustive de ce document sera fournie au cours des itérations ultérieures.

# 2 Architecture d’EndoMine

Décision	Motifs
Style architectural module	N.A.
Style architectural Composantes et Connecteurs	Utile pour décrire la communication entre les composantes.
Style architectural déploiement	Le style déploiement décrit où les composantes logicielles seront physiquement déployées. Ce diagramme inclut aussi les ports de communications nécessaires entre diverses machines.

## 2.1 Composantes d'EndoMine

### 2.1.1 Diagrammes des composants

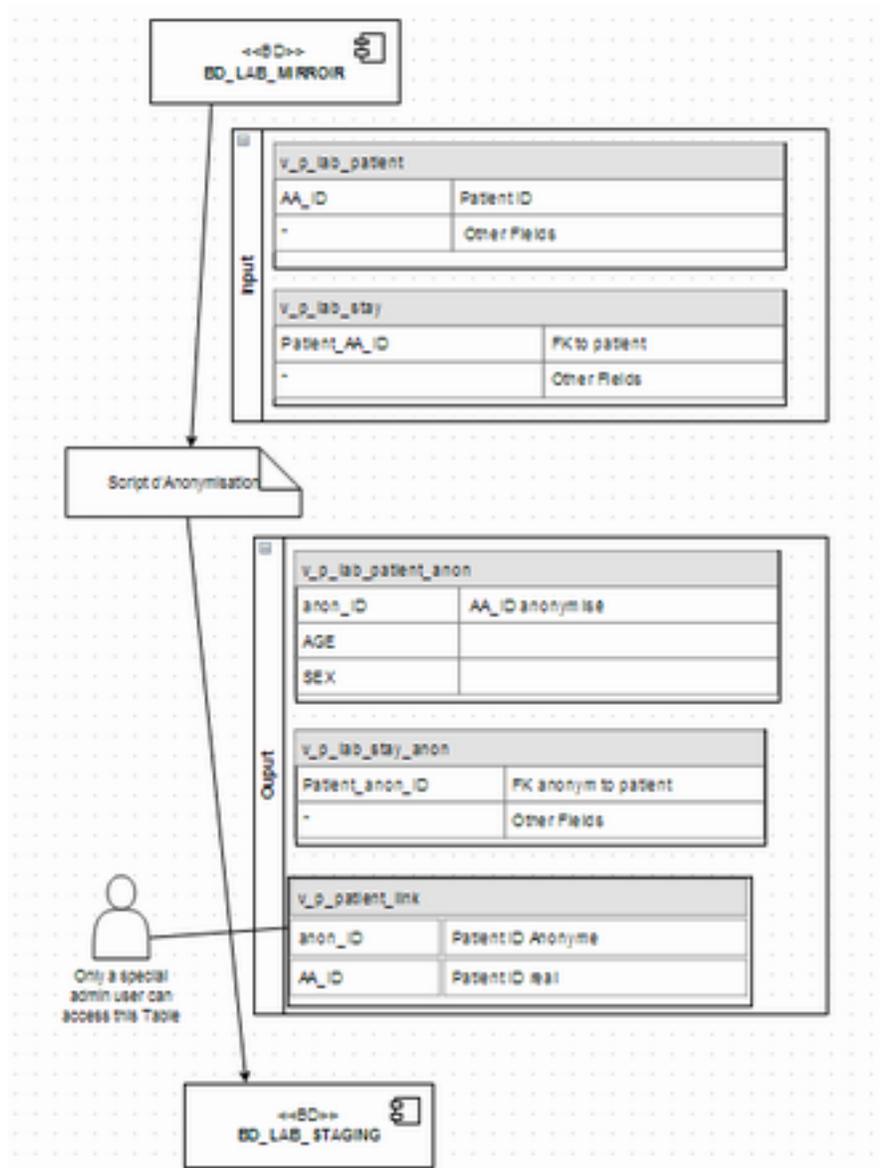


Figure 1: Diagramme des composants

### 2.1.2 Processus d'anonymisation

Le script d'anonymisation est exécuté par un administrateur de base de données (DBA). La table **v\_p\_patient\_link** sert de lien entre les IDs des patients anonymisés et non anonymisés. Cette table serait seulement accessible à un utilisateur spécial de la BD de *staging* (présentement seul Dr. Eintracht y a accès). La table **v\_p\_patient\_link** n'est pas importée dans le système distribué de forage de données.

## 2.2 Déploiement d'EndoMine

### 2.2.1 Diagrammes de déploiement

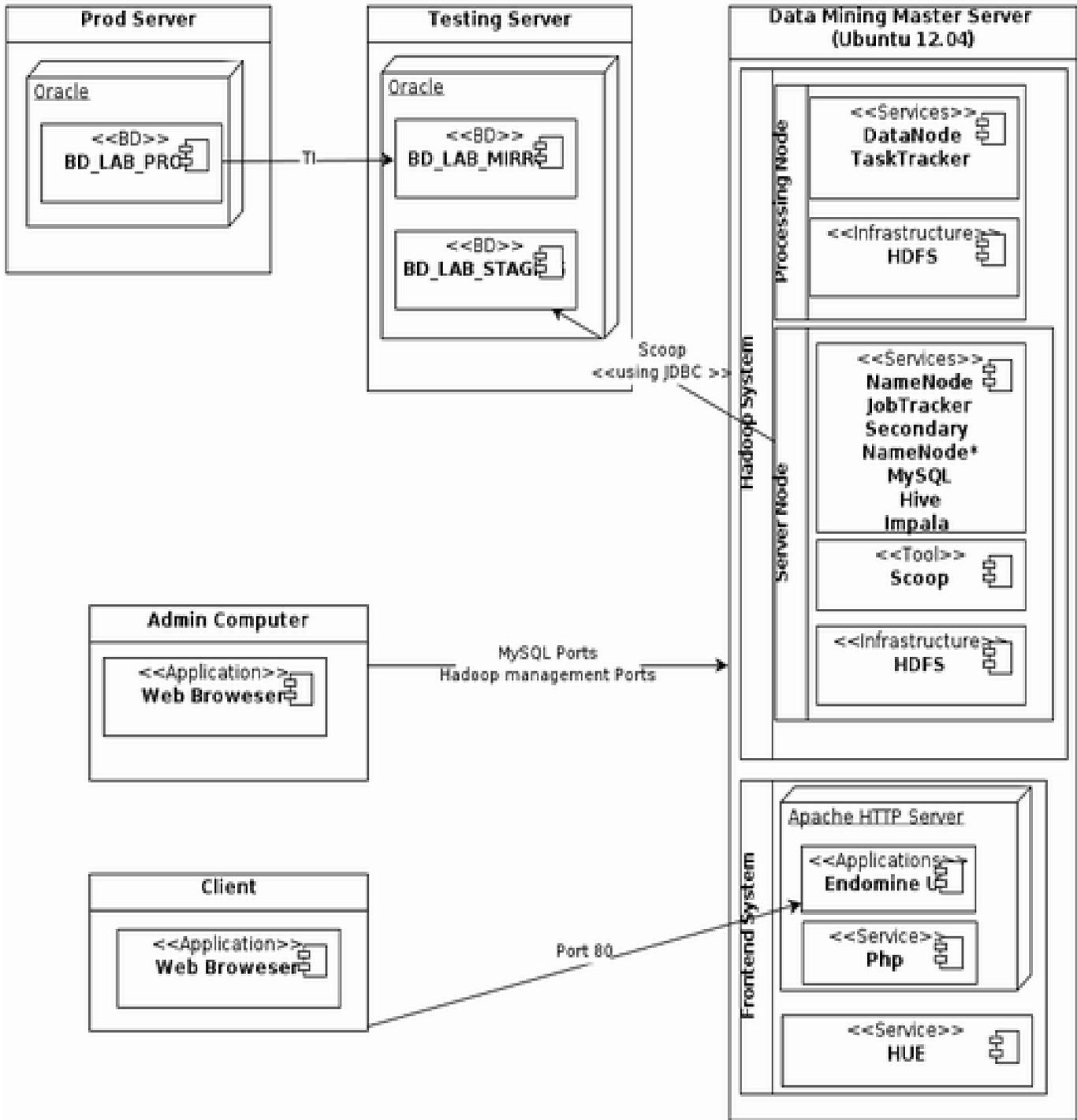


Figure 2 : Vue globale du déploiement du système

**Table 1. Description de déploiement.**

	<b>Éléments</b>	<b>Description</b>
DO1	Data Mining Master Server	Serveur fourni par l'ÉTS. Va contenir virtuellement tout au début.
DO2	Admin Computer	Ordinateur servant à faire de l'administration du serveur de l'ÉTS. C'est la seule machine qui doit avoir accès aux ports d'administration de Hadoop:  <i>Web UIs: 50070,50075,50090,50105,50030,50060</i> <sup>(1)</sup> Sert à voir l'activité des différentes tâches. Lors d'un problème, l'administrateur peut se connecter au serveur de l'ÉTS par SSH (Port <b>22</b> ). <i>Impala: 25000, 25010</i> <sup>(2)</sup> <i>Ajouter d'autres ports</i> <sup>(3)</sup> <i>au besoin.</i>  <i>MySQL Port : 3306</i>
DO3	Testing Server	Serveur de staging. Contient la copie de la prod (miroir) ainsi que la base de données de staging. Cette dernière est accédée par le Data Mining Master Server à travers JDBC sur <b>port XXXX</b> .
DO4	Client	Ordinateur du client utilisant le QueryBuilder. C'est la seule machine qui doit accéder au <b>port 80</b> du Data Mining Master Server.
DO5	Endomine UI	Application développée servant de QueryBuilder pour l'utilisateur. Accessible à travers un navigateur web.
D06	Prod Server	Contient la BD de LAB de production. C'est l'équipe de TI qui décide comment transférer les données de cette BD vers le <i>Testing Server</i> .

<sup>1</sup> Cloudera. (2009). *Hadoop Default Ports Quick Reference*. Consulté le 31 octobre 2012, à <http://blog.cloudera.com/blog/2009/08/hadoop-default-ports-quick-reference/>

<sup>2</sup> Cloudera. (2012). Installing and using Cloudera Impala. <https://ccp.cloudera.com/download/attachments/22152027/Impala-1.0-Beta-Installing-and-Using-Cloudera-Impala.pdf>

<sup>3</sup> Cloudera. (2012). *Configuring Ports for CDH4*. Consulté le 31 octobre 2012, à <https://ccp.cloudera.com/display/CDH4DOC/Configuring+Ports+for+CDH4>

## **3 Architecture Hadoop**

À venir.

### **3.1 Hive**

### **3.2 HDFS**

### **3.3 Impala**

## **Bibliographie**

À venir.

# Annexe E

## Compte-rendus de réunions

**Date:**

Jeudi, le 14 septembre 2012, à 12:00

**Lieu:**

JGH, salle de conférence du département d'endocrinologie, local E-104

**Ordre du jour:**

- Évaluer la faisabilité d'un projet de data mining au JGH.

**Participants:**

- Dr. Mark Trifiro
- Dr. Shaun Eintrach
- Chris Polly Kandriotist
- Dr. Brent Richards
- Hoi Yun Oriana Yu
- Dr. Alain April
- Fodil Belghait
- David Lauzon

**Rédacteur:**

David Lauzon

**Résumé des échanges:**

- Dans le cadre de son doctorat avec le Dr. Richards, Oriana veut établir une corrélation entre le cancer de la prostate et le diabète, à l'aide des données de résultats que possède le département de biochimie médicale. Autrement, il en coûterait 50 millions \$ pour réaliser une telle recherche.
- Shaun est en mesure de fournir directement une extraction des données pour Oriana.

## **Plan d'actions:**

- Il reste à déterminer si c'est le besoin du Dr. Trifiro ou du Dr. Shaun qui seront livrés en premier.

**Date:**

Mardi, le 18 septembre 2012, à 12:00

**Lieu:**

JGH, salle de conférence du département d'endocrinologie, local E-104

**Ordre du jour:**

- Lancer officiellement le projet
- Réviser le plan de projet afin d'obtenir l'autorisation de procéder

**Participants:**

- Dr. Mark Trifiro
- Dr. Shaun Eintracht
- Chris Polly Kandriotis
- Dr. Alain April
- Anton Zakharov
- David Lauzon

**Rédacteur:**

David Lauzon

**Résumé des échanges:**

- Anton est introduit dans le projet.
- Le plan de projet est conforme aux attentes du JGH.
- Le focus de la première itération du projet sera sur les besoins de Shaun.
- Le Dr. Sami Suissa du département d'épidémiologie pourrait être éventuellement intéressé par le projet et donc il pourrait contribuer ses attentes et visions.

- Stéphane Benhamou ayant réalisé plusieurs solutions techniques avec le département d'épidémiologie pourrait contribuer son expertise sur les travaux de data mining effectués au JGH.

### **Plan d'actions:**

- Anton et David rencontreront Shaun sous une base régulière dans le but d'évaluer les besoins de Shaun.
- Réalisation du document de Vision
- Recherche de solutions “*big data*” gratuites pouvant être utilisées dans le cadre du projet.
- Inviter le Dr. Sami Suissa, et Stéphane Benhamou aux prochaines réunions.

**Date:**

Jeudi, le 4 octobre 2012, à 13:00

**Lieu:**

JGH, salle de conférence du département d'endocrinologie, local E-104

**Ordre du jour:**

- Comprendre comment et dans quels buts les utilisateurs du département d'endocrinologie veulent utiliser le système de data mining.
- Solliciter une première rétro-action sur une version préliminaire du document de Vision.

**Participants:**

- Dr. Mark Trifiro
- Stéphane Benhamou
- Anton Zakharov
- David Lauzon

**Rédacteur:**

David Lauzon

**Résumé des échanges:**

- Nouvelle personne dans le projet: Stéphane est un spécialiste en IT pour le département d'épidémiologie, et a monté plusieurs solutions (OLAP / B.I.) de data mining au JGH. Son implication dans le projet dépend de la décision du Dr. Suissa. S'il embarque dans le projet, il pourrait faciliter la communication entre les médecins et notre équipe.
- Une ébauche du document de vision a été présenté, et les commentaires ont confirmé que le projet est dans la bonne voie.
- La question principale que Mark aimerait résoudre est : "*Does results from Test A correlates with results from Test B ?*".

- Mark a mentionné un intérêt d'avoir une relation entre les patients et l'utilisation de médicaments, mais cela est hors de portée pour l'instant.

### **Plan d'actions:**

- Compléter le document de Vision

**Date:**

Jeudi, le 11 octobre 2012, à 10:00

**Lieu:**

JGH, salle de conférence du département d'endocrinologie, local E-104

**Ordre du jour:**

- Présenter le document de Vision (0.9.1) aux parties prenantes.
- Récolter la rétroaction et modifier le document de Vision, si nécessaire.

**Participants:**

- Dr. Mark Trifiro
- Dr. Shaun Eintracht
- Chris Polly Kandriotis
- Anton Zakharov
- David Lauzon

**Rédacteur:**

David Lauzon

**Résumé des échanges:**

- Petits changements : la BD de *staging* sera une copie complète de la BD de production, et cela devrait être complété d'ici quelques jours. Dr. Eintracht effectuera le suivi avec Luigi / Pino (en charge du IT du département de Biochimie).
- Le processus d'anonymisation/désanonymisation des données devront être produits par l'équipe de l'ÉTS.
- FEA06 (synchronisation automatique entre staging et data mining) ne sera plus automatique, mais manuellement exécuté par Dr. Eintracht lorsque nécessaire. La raison est que les mises à jours de la table des diagnostiques lui est fournie par une extraction manuelle en Excel réalisé

par un autre département.

- FEA09 (exportation) n'est pas requise immédiatement.
- Nouvelle fonctionnalité demandée : Filtrage des données tel que AutoFilter dans Excel.
- Nouvelle fonctionnalité demandée : Pouvoir voir les valeurs distinctes d'une colonne.
- EndoMine devrait permettre de réaliser le processus de recherche (incluant sous-recherche(s) et autres raffinement de recherche) en quelques heures au plus.
- Une nouvelle table contenant la liste des coûts par test sera ajoutée au système. Cette table représente combien chaque l'exécution de chaque catégorie de test coûte à l'hôpital.
- Enlever le Dr. Suissa de la liste des stakeholders du projet
- Dr. Mac Namara veut que la fonctionnalité d'exportation (FEA09) ne soit disponible pour les utilisateurs dont le Dr. Eintracht a approuvé l'accès. Les utilisateurs pourraient quand même voir les résultats, mais pas les télécharger.

### **Plan d'actions:**

- Lundi 15 / mardi 16 octobre, Chris s'occupera de la virtualization du serveur fourni pas l'ÉTS et configurera:
  - Une VM pour la BD miroir (Windows 2008)
  - Une VM pour EndoMine (Ubuntu 12.04)
- À partir des caractéristiques identifiées (FEA01 - FEA12), un document de spécifications d'exigences logicielles (S.R.S.) sera élaboré.
- La date de présentation prévue du SRS est le 1er octobre 2012 (sujet à changement).
- À noter: Anton Zakharov et David Lauzon seront indisponibles entre le 21 et le 27 octobre.

**Date:**

Jeudi, le 18 octobre 2012, à 17:00

**Lieu:**

École de Technologie Supérieure, département de génie logiciel et des T.I., local A-3085

**Ordre du jour:**

- Faire le point sur l'orientation du projet
- Mieux définir la portée du projet
- Discuter des solutions techniques permettant de résoudre notre problème de forage de données

**Participants:**

- Anton Zakharov
- David Lauzon
- Christian Desrosiers
- Alain April
- Fodil Belghait

**Rédacteur:**

Anton Zakharov

**Résumé des échanges:**

- La portée du projet présente est trop large pour le temps qui reste. La décision à été prise de construire un système de forage de données axé à optimiser certaines requêtes essentielles du Dr. Shaun. L'entrée des données et les optimisations pour d'autres requêtes que les requêtes type pourraient être faites durant la prochaine itération.
- La discussion sur le forage nous a fait considérer le problème sous l'angle du BI. Mais nous ne savons pas encore si nous pourrions utiliser une de leurs techniques.
- Une interface minimaliste pour faire des recherche devrait être réalisée.

- Étant donné que faire le projet «*from scratch*» est devenu irréaliste à la suite des discussions, M. Desrosiers nous avait proposé d'utiliser Pentaho, un *framework* « *open source* », qui peut faire de l'ETL, forage de données et BI.
- Nous avons discuté de la façon donc on pourrait obtenir les données rapidement de l'hôpital. Alain a dit qu'il va nous aider la dessus.
- Nous devons faire attention à ne pas nous embarquer dans les politiques de l'hôpital. Exemple données de diagnostiques.

### **Plan d'actions:**

- Examiner Pentaho et décider si ça vaut la peine de l'utiliser.
- Rencontrer Dr. Shaun pour pouvoir cibler les requêtes.
- Préparer et discuter du SRS.

**Date:**

Lundi, le 29 octobre 2012, à 15:30

**Lieu:**

JGH, département de biochimie médicale, local D-123

**Ordre du jour:**

- Cibler les types requêtes que le Dr. Shaun exécute le plus souvent
- Déterminer quand la BD miroir sera accessible
- Déterminer quand le serveur de data mining sera virtualisé et accessible

**Participants:**

- Anton Zakharov
- David Lauzon
- Dr. Shaun Eintracht

**Rédacteur:**

David Lauzon

**Résumé des échanges:**

- Shaun mentionne que le JGH possède déjà un serveur de test pour leur BD de production Oracle. Donc le service des T.I. du JGH pourra y installer une BD miroir sans coûts de licence Oracle supplémentaire. Ils s'occuperont de la synchronisation avec la BD de production. Ce qui libère le serveur de data mining pour notre projet.
- Le serveur de data mining n'a toujours pas été virtualisé. Il subsiste un problème technique dont Shaun a oublié la cause.
- Shaun estime le nombre de spécimens traités par jour à 2000-3000; soit environ 20,000 résultats de tests par jour.

- Lors de ces requêtes, Shaun entre toujours une date (sauf quand il s'agit d'une recherche sur toutes les données).
  - Les différents champs de date sont les suivants:
    - OrderDate: lorsque la commande de tests est inscrite dans le système.
    - CollectDate: lorsque l'échantillon (e.g. *Sample*) du specimen du patient est récolté par un employé.
    - ReceiveDate: lorsque l'échantillon est reçu par une des machines du laboratoire (e.g. *analyzers*)
    - TestDate: lorsqu'un équipement secondaire (e.g. *Instrument*) calcule un résultat de test. Habituellement cela ne prend quelques secondes.
    - VerifiedDate: lorsqu'un humain vérifie que le résultat de test produit par l'instrument est valide.
  - Deux types de processus de résultat de test sont possibles (différences en gras) :
    - Test Center: **OrderDate**, **CollectDate**, ReceiveDate, TestDate, VerifiedDate
    - Wards (e.g. dans l'hôpital): **CollectDate**, **OrderDate**, ReceiveDate, TestDate, VerifiedDate
  - 90% des résultats de test sont traités dans la même journée, mais Shaun doit quand même pouvoir rechercher tous les différents champs de date.
  - Dans le système actuel, pour des raisons de performance Shaun fait ses requêtes sur le TestOrderID car celui-ci peut être converti en date.
  - Dans le système idéal, Shaun voudrait utiliser la ReceiveDate comme date principale pour effectuer les recherches.
- 5 requêtes SQL générées par le *Query Builder* de Shaun pour préparer les cas d'utilisations du SRS.
- Une requête relativement simple sans filtre a été chronométrée à 2m50s pour extraire seulement 14000 rangées.

- La fonction SQL écrite par Shaun pour convertir le TestOrderID en date a été fourni. Cependant, il ne fonctionne que pour les dates récentes.
- Excellente nouvelle: un TestResult peut être associé à VPSpecimen avec le TestOrderID, TestID, et SpecimenType. Cela devrait permettre de dénormaliser complètement les 4 tables principales.
- Les groupes de test ont été expliqués:
  - Chaque résultat de test est associé à un GroupTestID et un TestID, et ces deux champs réfèrent à la colonne ID de la table Test.
  - Lorsque le ID de la table Test représente un groupe, il y'a un enregistrement GroupTest est créé ayant le même TestID (cependant les AA\_ID ne correspondent pas).
  - Chaque GroupTest peut être composé jusqu'à 40 tests, qui peuvent être à leur tour des GroupTests. Il s'agit donc d'une structure en arbre.
- Pour ce qui est des tables de grandes dimensions (patient, séjour, commande, spécimen, résultat), Shaun n'aura pas besoin de faire des “*self-join*” d'une table à elle-même. Cela simplifie l'application que les autres tables sont petites et pourront être contenues en mémoire.

## Plan d'actions:

- Shaun va effectuer le suivi pour s'assurer que la BD miroir sera disponible d'ici le prochain rendez-vous du 1<sup>er</sup> novembre.
- Shaun va effectuer les démarches avec le département de TI du JGH, pour qu'Anton et David puisse ramener le serveur de data mining après la rencontre du 1er novembre. L'installation de l'éco-système Hadoop sera plus facile à l'extérieur du JGH (le serveur ne contiendra aucune donnée confidentielle).
- Réaliser le SRS en visant l'utilisation de Hadoop Impala et d'un *QueryBuilder opensource*.

**Date:**

Jeudi, le 1<sup>er</sup> novembre, à 10:00

**Lieu:**

JGH, Salle de conférence du département de biochimie médicale, local D-127

**Ordre du jour:**

- Présenter le SRS et la première version du document d'architecture.

**Participants:**

- Anton Zakharov
- David Lauzon
- Dr. Elizabeth Mac Namara
- Dr. Shaun Eintracht
- Chris Polykandriotis
- Stephane Benhamou

**Rédacteur:**

Anton Zakharov

**Résumé des échanges:**

- Dr. Elizabeth Mac Namara avait soulevé plusieurs points dont nous devons tenir compte:
  1. La façon de collecter, mesurer et/ou gérer les résultats des tests ont évolué à travers le temps. Ce qui fait que les mêmes données dans les mêmes colonnes de la table **v\_p\_test\_results** peuvent dire différentes choses dépendamment de la date où ils ont été entrées. Ces données devraient être flagés dans le système de forage de données. Limiter l'accès à ces données (raison de ne pas avoir d'utilisateurs multiples).
  2. L'accès aux données patients est régi par le comité d'éthique qui prends du temps. Faut éviter de mettre des données qui peut requérir leurs services.

3. Un patient peut avoir plusieurs IDs. Une proposition avait été proposée de garder juste les patients qui ont un numéro de la RAMQ.
- Stéphane Benhamou avait soulevé les points suivants :
    1. Pertinence d'utiliser Hadoop pour l'utilisation d'une si petite quantité de données.
    2. Question de maintenance du cluster BigData.
    3. Question de sécurité des données du cluster. Nous lui avons montré le document d'architecture, mais nous n'avons pas vraiment eu de feedback.
    4. Le fait que les données devraient être nettoyées si on veut utiliser la solution BI. (ce qui prendrait 2 ans selon Dr. Mac Namara).
  - Anton et David ont fait un résumé de la partie du document de vision que Stéphane n'avait pas lu. Mais nous n'avons pas pu présenter le SRS.

### **Plan d'actions:**

- Documenter la pertinence d'utiliser Hadoop.
- Ajouter un champ dans la grande table pour indiquer que les données ne sont pas propres.
- Nettoyer le document de Vision (acteurs, page couverture, besoins).

**Date:**

Jeudi , le 1<sup>er</sup> novembre, à 17:00

**Lieu:**

École de Technologie Supérieure, département de génie logiciel et des T.I., local A-3085

**Ordre du jour:**

- Présenter le SRS.

**Participants:**

- Anton Zakharov
- David Lauzon
- Christian Desrosiers
- Alain April

**Rédacteur:**

Anton Zakharov

**Résumé des échanges:**

- Alain à suggéré d'ajouter des contraintes dans le prototype pour se protéger.

**Plan d'actions:**

- Ajouter des contraintes aux SRS.
- Faire approuver et signer les documents par le client, pour pouvoir débiter l'implémentation.
- Commencer à travailler sur la partie concrète du projet (Installer Hadoop + Impala sur le serveur) .

**Date:**

Mardi , le 6 novembre, à 15:00

**Lieu:**

JGH, Département de biochimie médicale, local D-123

**Ordre du jour:**

- Faire signer le document de Vision et le SRS par le département de biochimie médicale.
- Éclaircir certains détails

**Participants:**

- Dr. Elizabeth Mac Namara
- Dr. Shaun Eintracht
- Anton Zakharov
- David Lauzon

**Rédacteur:**

David Lauzon

**Résumé des échanges:**

- Dr. Mac Namara et Dr. Eintracht ont validé et signer les documents de Vision et SRS.
- “Pluri-identité” d'un même patient (c.-à-d. un même patient ayant plusieurs PatientID) :
  - Shaun explique que d'habitude les fusions sont faites manuellement directement dans la BD de production.
  - Une solution pouvant faciliter la tâche serait d'ajouter un champs supplémentaire “SamePatientID” dans EndoMine, que Shaun pourrait populer au besoin (livraison non garantie dans le cadre de ce PFE).
- La compagnie qui gère l'implantation du logiciel SoftLab LIS au JGH est MediSolution .

- Shaun nous fait un survol du CMS (système de gestion de contenu) qu'il a monté avec la plateforme Camtesia (ASP).
  - Tous les employés qui utilisent le laboratoire du département de médecine diagnostique doivent passer un entraînement (video, pdf, entre autres) sur ce CMS.
  - Le CMS permet de tracer quels employés ont lu quels consignes / messages / etc.
  - Lorsqu'il y'a un changement (nouvel appareil, changement de méthodologie dans la prise d'un test en particulier), Shaun ajoute un document dans le CMS. Ce document doit être lu par tous les employés utilisant les machines affectés.
  - Le CMS contient un “*mapping*” des tests et groupe de tests, et permet de rechercher plus facilement la hiérarchie d'un groupe de test qu'avec la BD Oracle / Access. Aussi, un descriptif textuel est disponible pour chaque test.
- Données malpropres :
  - Exemple :
    1. Un test (vs\_lab\_test) performé à donné une valeur de 140 pour un patient (vp\_lab\_test\_result). À ce moment les valeurs de référence (R.R. Ou “*Reference Range*”) étaient entre 135 et 145.
    2. Par la suite, des changements sont apportés à la méthodologie de la prise de ce test (changement d'équipement, calibration, changement de paramètres, etc.).
      - **Note: la plupart du temps un nouvel enregistrement est créé dans la table vs\_lab\_test avec le même Test\_ID mais un Test\_AA\_ID différent. Le champs “active” du “vieux” test est mis à FAUX, et celui du “nouveau” test à VRAI pour distinguer le test actif. Cependant, il est possible que le test (vs\_lab\_test) ait été modifié directement – donc perte potentielle d'information (à vérifier).**
    3. Suite au changement la valeur de l'ancien test devrait valoir 141 au lieu de 140. Et les R.R. sont devenus entre 137 et 148.
  - Dr Mac Namara ne veut pas nettoyer toutes les données d'un seul coup à cause de l'effort

humain que cela prendrait (environ 2 ans). Plutôt, elle veut pouvoir les modifier selon la demande; par exemple un test à la fois.

- Certaines données, probablement les plus anciennes, ne pourront jamais être nettoyées car il est possible que le changement ne soit plus retraceable.
- Solution proposée (livraison non garantie dans le cadre de ce PFE):
  - Ajout d'une nouvelle table: vs\_lab\_test\_calibration
  - Champs: TestID, StartDate, EndDate, **ValueFactor**, **RRLow**, **RRHigh**, Dirty
  - La donnée enregistrée originalement dans vp\_lab\_test\_result n'est jamais modifiée. Pour la recalibrer suffit de la multiplier par le ValueFactor (valeur par défaut = 1.0) lors de la requête. Par exemple,  $150 * 1.1 = 165$ .
  - RRLow et RRHigh contiendrait les valeurs de référence correspondante pour le test et la date à laquelle le test a été effectué (spécifiée).
  - Dirty = 1 signifie que les résultats du test ne sont pas propres et/ou il n'est pas possible de les uniformiser.
  - Si StartDate vaut NULL, alors cela équivaut à la date de la première donnée entrée dans le système.
  - Si EndDate vaut NULL, alors cela équivaut à la date de la dernière donnée entrée dans le système.
  - Si aucun enregistrement existe dans vs\_lab\_test\_calibration pour un TestID et une date spécifiée, alors une valeur de 1.0 est assumée pour le ValueFactor et on assume que la valeur est malpropre (dirty = 0).
  - Extensions possibles :
    - Champs pour commenter le changement.
    - Champs pour indiquer l'auteur du changement.
    - Champs pour indiquer le type d'opération à faire avec ValueFactor (ex: MUL pour multiplication)

- Shaun a fourni les données pour les tables vs\_lab\_clinic, vs\_lab\_doctor (sans le numéro de téléphone et l'adresse), et vs\_lab\_workstation.

## Plan d'actions:

- Shaun va construire une requête plus complexe que celles fournies dans le but de comparer l'ancien système avec le nouveau.
- Première implantation du système prévue le jeudi 15 novembre.
  - Shaun va mettre de la pression sur le IT pour s'assurer d'avoir la BD miroir prête, et la BD de staging créée.
  - On va amener le serveur au JGH et tester le réseautage (adresse IPs, etc.), la connexion à Oracle, et la connexion depuis notre laptop au serveur.
  - On va charger les données anonymisées dans le serveur d'Alain.
- Dénormalisation des 4 grandes tables : la jointure sur le specimen\_type et le workstation\_id génèrent encore quelques (environ 200 sur 65000 résultats de test) duplicats. Il reste encore du travail à faire la dessus.
- Dr. Mac Namara nous recommande de lui faire remplir un formulaire d'évaluation de nos performances d'analyse de besoin, et suggère que cette évaluation soit prise en compte dans notre PFE.

## Annexe F

# Rapport de résolution des incidents

# **Rapport de résolution des incidents**

## **EndoMine**

Hôpital Général Juif

version 0.1

**Anton Zakharov**  
**David Lauzon**

**École de technologie supérieure**  
**le 15 décembre 2012**

# Resolution des incidents

## Tables des matières

[Introduction](#)

[Audience](#)

[Objectif](#)

[Probleme 1 : Attempted to generate class with no columns!](#)

[Probleme 2 : OOME: Direct buffer memory](#)

[Probleme 3 : Could not find any valid local directory for output/spill1.out](#)

[Probleme 4 : java.io.IOException: Xceiver count 4097 exceeds the limit of concurrent xcievers: 4096](#)

[Probleme 5 : DataXceiver error processing WRITE\\_BLOCK operation \(SocketTimeoutException\)](#)

## **Introduction**

Ce rapport résume quelques problèmes qui ont été rencontrés lors de l'implémentation du prototype, et comment ils ont été résolus.

## **Audience**

Les éventuels partis qui seront amenés à poursuivre le développement du produit et/ou à le maintenir.

## **Objectif**

L'objectif de ce document est de familiariser les lecteurs avec la cause et la résolution de ces problèmes, afin de leur faire économiser temps et énergie.

# Probleme 1 : Attempted to generate class with no columns!

## Technologies :

Sqoop, Oracle

## Symptomes :

```
+ sudo -u endomine sqoop import --connect jdbc:oracle:thin:@192.168.1.118:1521:xe --username anton
--password thepassword --table vs_lab_test_group --hive-import --hive-table vslabgroup7
12/11/19 23:38:36 WARN tool.BaseSqoopTool: Setting your password on the command-line is insecure.
Consider using -P instead.
12/11/19 23:38:36 INFO tool.BaseSqoopTool: Using Hive-specific delimiters for output. You can
override
12/11/19 23:38:36 INFO tool.BaseSqoopTool: delimiters with --fields-terminated-by, etc.
12/11/19 23:38:36 INFO manager.SqlManager: Using default fetchSize of 1000
12/11/19 23:38:36 INFO tool.CodeGenTool: Beginning code generation
12/11/19 23:38:37 INFO manager.OracleManager: Time zone has been set to GMT
12/11/19 23:38:37 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM
VS_LAB_TEST_GROUP t WHERE 1=0
12/11/19 23:38:37 ERROR tool.ImportTool: Imported Failed: Attempted to generate class with no
columns!
```

## Diagnostique :

While working with Oracle you may encounter problems when Sqoop can not figure out column names. This happens because the catalog queries that Sqoop uses for Oracle expect the correct case to be specified for the user name and table name.

## Resolution :

1. Specify the user name, which Sqoop is connecting as, in upper case (unless it was created with mixed/lower case within quotes).
2. Specify the table name, which you are working with, in upper case (unless it was created with mixed/lower case within quotes).

Ex:

```
sudo -u endomine sqoop import --connect jdbc:oracle:thin:@192.168.1.118:1521:xe --username ANTON --
password THEPASSWORD --table VS_LAB_TEST_GROUP --hive-import --hive-table VSLABGROUP7
```

## References :

<http://sqoop.apache.org/docs/1.4.0-incubating/SqoopUserGuide.html#id1774574>

## Probleme 2 : OOME: Direct buffer memory

### Technologies :

MapReduce, Hive, Snappy

### Symptomes :

The following error appears in TaskTracker's log file during the reduce phase :

```
2012-11-29 05:11:08,628 FATAL org.apache.hadoop.mapred.Child: Error running child :
java.lang.OutOfMemoryError: Direct buffer memory
    at java.nio.Bits.reserveMemory(Bits.java:632)
    at java.nio.DirectByteBuffer.<init>(DirectByteBuffer.java:97)
    at java.nio.ByteBuffer.allocateDirect(ByteBuffer.java:288)
    at org.apache.hadoop.io.compress.snappy.SnappyCompressor.<init>(SnappyCompressor.java:81)
    at org.apache.hadoop.io.compress.SnappyCodec.createCompressor(SnappyCodec.java:142)
    at org.apache.hadoop.io.compress.CodecPool.getCompressor(CodecPool.java:106)
    at org.apache.hadoop.io.compress.CodecPool.getCompressor(CodecPool.java:118)
    at org.apache.hadoop.io.SequenceFile$Writer.init(SequenceFile.java:1169)
    at org.apache.hadoop.io.SequenceFile$Writer.<init>(SequenceFile.java:1080)
    at org.apache.hadoop.io.SequenceFile$BlockCompressWriter.<init>(SequenceFile.java:1400)
    at org.apache.hadoop.io.SequenceFile.createWriter(SequenceFile.java:274)
    at org.apache.hadoop.io.SequenceFile.createWriter(SequenceFile.java:369)
    at org.apache.hadoop.hive.ql.exec.Utilities.createSequenceWriter(Utilities.java:969)
    at
org.apache.hadoop.hive.ql.io.HiveSequenceFileOutputFormat.getHiveRecordWriter(HiveSequenceFileOutputFormat.java:64)
    at
org.apache.hadoop.hive.ql.io.HiveFileFormatUtils.getRecordWriter(HiveFileFormatUtils.java:246)
    at
org.apache.hadoop.hive.ql.io.HiveFileFormatUtils.getHiveRecordWriter(HiveFileFormatUtils.java:234)
    at
org.apache.hadoop.hive.ql.exec.FileSinkOperator.createBucketFiles(FileSinkOperator.java:477)
    at org.apache.hadoop.hive.ql.exec.FileSinkOperator.getDynOutPaths(FileSinkOperator.java:629)
    at org.apache.hadoop.hive.ql.exec.FileSinkOperator.processOp(FileSinkOperator.java:563)
    at org.apache.hadoop.hive.ql.exec.Operator.process(Operator.java:471)
    at org.apache.hadoop.hive.ql.exec.Operator.forward(Operator.java:762)
    at org.apache.hadoop.hive.ql.exec.SelectOperator.processOp(SelectOperator.java:84)
    at org.apache.hadoop.hive.ql.exec.Operator.process(Operator.java:471)
    at org.apache.hadoop.hive.ql.exec.Operator.forward(Operator.java:762)
    at org.apache.hadoop.hive.ql.exec.SelectOperator.processOp(SelectOperator.java:84)
    at org.apache.hadoop.hive.ql.exec.Operator.process(Operator.java:471)
    at org.apache.hadoop.hive.ql.exec.Operator.forward(Operator.java:762)
    at
org.apache.hadoop.hive.ql.exec.CommonJoinOperator.genUniqueJoinObject(CommonJoinOperator.java:742)
    at
org.apache.hadoop.hive.ql.exec.CommonJoinOperator.genUniqueJoinObject(CommonJoinOperator.java:745)
    at
org.apache.hadoop.hive.ql.exec.CommonJoinOperator.checkAndGenObject(CommonJoinOperator.java:852)
    at org.apache.hadoop.hive.ql.exec.JoinOperator.endGroup(JoinOperator.java:265)
    at org.apache.hadoop.hive.ql.exec.ExecReducer.reduce(ExecReducer.java:198)
    at org.apache.hadoop.mapred.ReduceTask.runOldReducer(ReduceTask.java:492)
    at org.apache.hadoop.mapred.ReduceTask.run(ReduceTask.java:436)
```

```
at org.apache.hadoop.mapred.Child$4.run(Child.java:268)
at java.security.AccessController.doPrivileged(Native Method)
at javax.security.auth.Subject.doAs(Subject.java:396)
at org.apache.hadoop.security.UserGroupInformation.doAs(UserGroupInformation.java:1332)
at org.apache.hadoop.mapred.Child.main(Child.java:262)
```

### Diagnostic :

Manque de mémoire lors de la compression.

### Resolution :

1. Augmenter la mémoire allouée à la JVM.
2. Sauvegarder le contenu de la pile lorsque que le noeud manque de mémoire.
3. mapred-site.xml

```
<property>
  <name>mapred.child.java.opts</name>
  <!-- Starts JVM with <Xms> MB, but can increase up to <Xmx> MB -->
  <value>-Xms2048m -Xmx2560m -XX:+HeapDumpOnOutOfMemoryError -XX:HeapDumpPath=/var/log/hadoop-
0.20-mapreduce/ </value>
</property>
```

### References :

<http://www.oracle.com/technetwork/java/javase/memleaks-137499.html#gdyrr>  
<https://issues.apache.org/jira/browse/HDFS-3600>

# Probleme 3 : Could not find any valid local directory for output/spill1.out

**Technologies :**  
MapReduce, Hive

## Symptomes :

The following error appears in TaskTracker's log file during a heavy map phase :

```
2012-11-30 01:33:49,636 WARN org.apache.hadoop.mapred.Child: Error running child
java.lang.RuntimeException: org.apache.hadoop.hive.ql.metadata.HiveException: Hive Runtime Error
while processing row
```

```
{ "aa_id":1.83130226E8, "test_date":2.0080605E7, "test_time":832.0, "verified_date":2.0080605E7, "verified_time":832.0, "qc_time_violate": "N", "bad_qc_result": "N", "report_to_his": "N", "not_callable_test": "N", "not_report_test": "N", "test_reported": "N", "hidden_result": "N", "abnormal_test_report": "N", "workshe eted_flag": "N", "downloaded_flag": "Y", "edited_flag": "N", "posted_flag": "Y", "verified_flag": "Y", "reported_flag": "Y", "tournaround_violat": "N", "reflex_test": "N", "result_not_changed": "N", "credited_flag": "N", "billed_flag": "N", "rerun_flag": "N", "spec_collected": "Y", "spec_received": "Y", "ref_lab": "N", "workf lag": "N", "panic_low": "N", "panic_high": "N", "abnormal_low": "Y", "abnormal_high": "N", "percent_delta": "N", "absolute_delta": "N", "absurd_low": "N", "absurd_high": "N", "calculated": "N", "workload_for_drg": "N", "workload_for_stats": "N", "d7_reported": "N", "d_reported": "N", "t_reported": "N", "f_reported": "Y", "baci_test": "N", "test_id": "MPV", "group_test_id": "CBC", "result": "9.7", "technik_id": "I/AUT", "reviewer_id": "I/AUT", "ordering_workstation_id": "STKS", "reflex_date": 0.0, "reflex_time": 0.0, "reflex_test_id": "null", "testing_workstation_id": "SIS", "priority": "R", "aflags4": "null", "order_sort": 10.0, "order_aa_id": 5999750.0, "specimen_type": "LHH" }
at org.apache.hadoop.hive.ql.exec.ExecMapper.map(ExecMapper.java:161)
at org.apache.hadoop.mapred.MapRunner.run(MapRunner.java:50)
at org.apache.hadoop.mapred.MapTask.runOldMapper(MapTask.java:393)
at org.apache.hadoop.mapred.MapTask.run(MapTask.java:327)
at org.apache.hadoop.mapred.Child$4.run(Child.java:268)
at java.security.AccessController.doPrivileged(Native Method)
at javax.security.auth.Subject.doAs(Subject.java:396)
at org.apache.hadoop.security.UserGroupInformation.doAs(UserGroupInformation.java:1332)
at org.apache.hadoop.mapred.Child.main(Child.java:262)
```

Caused by: org.apache.hadoop.hive.ql.metadata.HiveException: Hive Runtime Error while processing row

```
{ "aa_id":1.83130226E8, "test_date":2.0080605E7, "test_time":832.0, "verified_date":2.0080605E7, "verified_time":832.0, "qc_time_violate": "N", "bad_qc_result": "N", "report_to_his": "N", "not_callable_test": "N", "not_report_test": "N", "test_reported": "N", "hidden_result": "N", "abnormal_test_report": "N", "workshe eted_flag": "N", "downloaded_flag": "Y", "edited_flag": "N", "posted_flag": "Y", "verified_flag": "Y", "reported_flag": "Y", "tournaround_violat": "N", "reflex_test": "N", "result_not_changed": "N", "credited_flag": "N", "billed_flag": "N", "rerun_flag": "N", "spec_collected": "Y", "spec_received": "Y", "ref_lab": "N", "workf lag": "N", "panic_low": "N", "panic_high": "N", "abnormal_low": "Y", "abnormal_high": "N", "percent_delta": "N", "absolute_delta": "N", "absurd_low": "N", "absurd_high": "N", "calculated": "N", "workload_for_drg": "N", "workload_for_stats": "N", "d7_reported": "N", "d_reported": "N", "t_reported": "N", "f_reported": "Y", "baci_test": "N", "test_id": "MPV", "group_test_id": "CBC", "result": "9.7", "technik_id": "I/AUT", "reviewer_id": "I/AUT", "ordering_workstation_id": "STKS", "reflex_date": 0.0, "reflex_time": 0.0, "reflex_test_id": "null", "testing_workstation_id": "SIS", "priority": "R", "aflags4": "null", "order_sort": 10.0, "order_aa_id": 5999750.0, "specimen_type": "LHH" }
```

```

at org.apache.hadoop.hive.ql.exec.MapOperator.process (MapOperator.java:548)
at org.apache.hadoop.hive.ql.exec.ExecMapper.map (ExecMapper.java:143)
... 8 more
Caused by: org.apache.hadoop.hive.ql.metadata.HiveException: java.io.IOException: Spill failed
at org.apache.hadoop.hive.ql.exec.ReduceSinkOperator.processOp (ReduceSinkOperator.java:304)
at org.apache.hadoop.hive.ql.exec.Operator.process (Operator.java:471)
at org.apache.hadoop.hive.ql.exec.Operator.forward (Operator.java:762)
at org.apache.hadoop.hive.ql.exec.TableScanOperator.processOp (TableScanOperator.java:83)
at org.apache.hadoop.hive.ql.exec.Operator.process (Operator.java:471)
at org.apache.hadoop.hive.ql.exec.Operator.forward (Operator.java:762)
at org.apache.hadoop.hive.ql.exec.MapOperator.process (MapOperator.java:529)
... 9 more
Caused by: java.io.IOException: Spill failed
at org.apache.hadoop.mapred.MapTask$MapOutputBuffer.collect (MapTask.java:876)
at org.apache.hadoop.mapred.MapTask$OldOutputCollector.collect (MapTask.java:501)
at org.apache.hadoop.hive.ql.exec.ReduceSinkOperator.processOp (ReduceSinkOperator.java:289)
... 15 more
Caused by: org.apache.hadoop.util.DiskChecker$DiskErrorException: Could not find any valid local
directory for output/spill1.out
at
org.apache.hadoop.fs.LocalDirAllocator$AllocatorPerContext.getLocalPathForWrite (LocalDirAllocator.j
ava:398)
at org.apache.hadoop.fs.LocalDirAllocator.getLocalPathForWrite (LocalDirAllocator.java:150)
at org.apache.hadoop.fs.LocalDirAllocator.getLocalPathForWrite (LocalDirAllocator.java:131)
at org.apache.hadoop.mapred.MapOutputFile.getSpillFileForWrite (MapOutputFile.java:121)
at org.apache.hadoop.mapred.MapTask$MapOutputBuffer.sortAndSpill (MapTask.java:1237)
at org.apache.hadoop.mapred.MapTask$MapOutputBuffer.access$1800 (MapTask.java:702)
at org.apache.hadoop.mapred.MapTask$MapOutputBuffer$SpillThread.run (MapTask.java:1189)
2012-11-30 01:33:49,644 INFO org.apache.hadoop.mapred.Task: Runnning cleanup for the task

```

## Diagnosticque :

L'une des tâche a manqué de RAM, et lorsqu'elle a voulu décharger son contenu sur le disque (e.g. *spill*), le repertoire temporaire sur le disque utilisé par les jobs mapreduce était plein.

## Resolution :

1. Augmenter la mémoire allouée à la JVM.
2. mapred-site.xml

```

<property>
  <name> ... todo
</property>

```

## References :

<http://stackoverflow.com/questions/11602074/hadoop-job-runs-okay-on-smaller-set-of-data-but-fails-with-large-dataset>  
[https://groups.google.com/a/cloudera.org/forum/?fromgroups=#!topic/cdh-user/syIM9j\\_QyI4](https://groups.google.com/a/cloudera.org/forum/?fromgroups=#!topic/cdh-user/syIM9j_QyI4)

# Probleme 4 : java.io.IOException: Xceiver count 4097 exceeds the limit of concurrent xcievers: 4096

**Technologies :**  
MapReduce, Hive

## Symptomes :

The following error appears in DataNode's log file during a heavy reduce phase :

```
2012-12-02 00:38:27,241 INFO org.apache.hadoop.hdfs.server.datanode.DataNode: Receiving block BP-1790420707-192.168.1.119-1353791420945:blk_-4351351169631282249_131066 src: /10.164.121.4:59330 dest: /10.164.121.4:50010
2012-12-02 00:38:27,445 WARN org.apache.hadoop.hdfs.server.datanode.DataNode: 008920a.intranet.jgh:50010:DataXceiverServer:
java.io.IOException: Xceiver count 4097 exceeds the limit of concurrent xcievers: 4096
    at org.apache.hadoop.hdfs.server.datanode.DataXceiverServer.run(DataXceiverServer.java:143)
    at java.lang.Thread.run(Thread.java:662)
2012-12-02 00:38:27,585 WARN org.apache.hadoop.hdfs.server.datanode.DataNode: 008920a.intranet.jgh:50010:DataXceiverServer:
java.io.IOException: Xceiver count 4097 exceeds the limit of concurrent xcievers: 4096
    at org.apache.hadoop.hdfs.server.datanode.DataXceiverServer.run(DataXceiverServer.java:143)
    at java.lang.Thread.run(Thread.java:662)
2012-12-02 00:38:27,594 WARN org.apache.hadoop.hdfs.server.datanode.DataNode: 008920a.intranet.jgh:50010:DataXceiverServer:
java.io.IOException: Xceiver count 4097 exceeds the limit of concurrent xcievers: 4096
    at org.apache.hadoop.hdfs.server.datanode.DataXceiverServer.run(DataXceiverServer.java:143)
    at java.lang.Thread.run(Thread.java:662)
2012-12-02 00:38:28,007 WARN org.apache.hadoop.hdfs.server.datanode.DataNode: 008920a.intranet.jgh:50010:DataXceiverServer:
java.io.IOException: Xceiver count 4097 exceeds the limit of concurrent xcievers: 4096
    at org.apache.hadoop.hdfs.server.datanode.DataXceiverServer.run(DataXceiverServer.java:143)
    at java.lang.Thread.run(Thread.java:662)
2012-12-02 00:38:28,091 WARN org.apache.hadoop.hdfs.server.datanode.DataNode: 008920a.intranet.jgh:50010:DataXceiverServer:
java.io.IOException: Xceiver count 4097 exceeds the limit of concurrent xcievers: 4096
    at org.apache.hadoop.hdfs.server.datanode.DataXceiverServer.run(DataXceiverServer.java:143)
    at java.lang.Thread.run(Thread.java:662)
2012-12-02 00:38:28,091 WARN org.apache.hadoop.hdfs.server.datanode.DataNode: 008920a.intranet.jgh:50010:DataXceiverServer:
java.io.IOException: Xceiver count 4097 exceeds the limit of concurrent xcievers: 4096
    at org.apache.hadoop.hdfs.server.datanode.DataXceiverServer.run(DataXceiverServer.java:143)
    at java.lang.Thread.run(Thread.java:662)
2012-12-02 00:39:26,474 INFO org.apache.hadoop.hdfs.server.datanode.DataNode: Exception in receiveBlock for BP-1790420707-192.168.1.119-1353791420945:blk_8921806853627617246_120546
java.net.SocketTimeoutException: 60000 millis timeout while waiting for channel to be ready for read. ch : java.nio.channels.SocketChannel[connected local=/10.164.121.4:50010 remote=/10.164.121.4:57557]
    at org.apache.hadoop.net.SocketIOWithTimeout.doIO(SocketIOWithTimeout.java:165)
    at org.apache.hadoop.net.SocketInputStream.read(SocketInputStream.java:154)
    at org.apache.hadoop.net.SocketInputStream.read(SocketInputStream.java:127)
    at java.io.FilterInputStream.read(FilterInputStream.java:116)
```

```
at java.io.BufferedInputStream.fill(BufferedInputStream.java:218)
at java.io.BufferedInputStream.read1(BufferedInputStream.java:258)
at java.io.BufferedInputStream.read(BufferedInputStream.java:317)
at java.io.DataInputStream.read(DataInputStream.java:132)
at org.apache.hadoop.io.IOUtils.readFully(IOUtils.java:192)

at org.apache.hadoop.hdfs.protocol.datatransfer.PacketReceiver.doReadFully(PacketReceiver.java:213)

at org.apache.hadoop.hdfs.protocol.datatransfer.PacketReceiver.doRead(PacketReceiver.java:134)

at
org.apache.hadoop.hdfs.protocol.datatransfer.PacketReceiver.receiveNextPacket(PacketReceiver.java:109)

at org.apache.hadoop.hdfs.server.datanode.BlockReceiver.receivePacket(BlockReceiver.java:414)

at org.apache.hadoop.hdfs.server.datanode.BlockReceiver.receiveBlock(BlockReceiver.java:635)
at org.apache.hadoop.hdfs.server.datanode.DataXceiver.writeBlock(DataXceiver.java:506)
at org.apache.hadoop.hdfs.protocol.datatransfer.Receiver.opWriteBlock(Receiver.java:98)
at org.apache.hadoop.hdfs.protocol.datatransfer.Receiver.processOp(Receiver.java:66)
at org.apache.hadoop.hdfs.server.datanode.DataXceiver.run(DataXceiver.java:219)
at java.lang.Thread.run(Thread.java:662)
```

### **Diagnostic :**

La requête causant problème tentait de créer BEAUCOUP de partitions automatiquement avec hive, cela demande aux datanodes d'avoir plus de 4096 fichiers d'ouverts.

Cependant, une "thread" doit être créé pour chaque fichier ouvert, et chaque "thread" java prend environ 1 MB de mémoire, donc il faut faire attention avec cette configuration.

Ce problème aurait moins de chance de survenir dans un cluster plus grand, car les fichiers ouverts seraient répartis à travers plusieurs datanodes.

### **Resolution :**

1. Ré-écrire la requête de sorte qu'elle ne crée pas plus que 4986 partitions.

### **References :**

<http://blog.cloudera.com/blog/2012/03/hbase-hadoop-xceivers/>

<https://ccp.cloudera.com/display/CDHDOC/HBase+Installation>

# Probleme 5 : DataXceiver error processing WRITE\_BLOCK operation (SocketTimeoutException)

## Technologies :

MapReduce, Hive

## Symptomes :

The following error appears in DataNode's log file during a heavy read and write operations (9 hours operation on a multiple sequential hive jobs) :

### DataNode Log

```
2012-12-14 06:55:27,060 INFO org.apache.hadoop.hdfs.server.datanode.DataNode: Receiving block BP-1790420707-192.168.1.119-1353791420945:blk_3826138436647305852_342344 src: /10.196.121.18:38696 dest: /10.196.121.18:50010
2012-12-14 06:56:27,140 INFO org.apache.hadoop.hdfs.server.datanode.DataNode: Exception in receiveBlock for BP-1790420707-192.168.1.119-1353791420945:blk_8120590457789607396_342342 java.net.SocketTimeoutException: 60000 millis timeout while waiting for channel to be ready for read. ch : java.nio.channels.SocketChannel[connected local=/10.196.121.18:50010 remote=/10.196.121.18:38694]
    at org.apache.hadoop.net.SocketIOWithTimeout.doIO(SocketIOWithTimeout.java:165)
    at org.apache.hadoop.net.SocketInputStream.read(SocketInputStream.java:154)
    at org.apache.hadoop.net.SocketInputStream.read(SocketInputStream.java:127)
    at java.io.FilterInputStream.read(FilterInputStream.java:116)
    at java.io.BufferedInputStream.fill(BufferedInputStream.java:218)
    at java.io.BufferedInputStream.read1(BufferedInputStream.java:258)
    at java.io.BufferedInputStream.read(BufferedInputStream.java:317)
    at java.io.DataInputStream.read(DataInputStream.java:132)
    at org.apache.hadoop.io.IOUtils.readFully(IOUtils.java:192)

at org.apache.hadoop.hdfs.protocol.datatransfer.PacketReceiver.doReadFully(PacketReceiver.java:213)

at org.apache.hadoop.hdfs.protocol.datatransfer.PacketReceiver.doRead(PacketReceiver.java:134)

at
org.apache.hadoop.hdfs.protocol.datatransfer.PacketReceiver.receiveNextPacket(PacketReceiver.java:109)

at org.apache.hadoop.hdfs.server.datanode.BlockReceiver.receivePacket(BlockReceiver.java:414)

at org.apache.hadoop.hdfs.server.datanode.BlockReceiver.receiveBlock(BlockReceiver.java:635)
    at org.apache.hadoop.hdfs.server.datanode.DataXceiver.writeBlock(DataXceiver.java:506)
    at org.apache.hadoop.hdfs.protocol.datatransfer.Receiver.opWriteBlock(Receiver.java:98)
    at org.apache.hadoop.hdfs.protocol.datatransfer.Receiver.processOp(Receiver.java:66)
    at org.apache.hadoop.hdfs.server.datanode.DataXceiver.run(DataXceiver.java:219)
    at java.lang.Thread.run(Thread.java:662)

[...]
2012-12-14 06:56:36,692 INFO org.apache.hadoop.hdfs.server.datanode.DataNode: PacketResponder: BP-1790420707-192.168.1.119-1353791420945:blk_8120590457789607396_342342, type=LAST_IN_PIPELINE, downstreams=0[:]: Thread is interrupted.
```

```

2012-12-14 06:56:37,076 INFO org.apache.hadoop.hdfs.server.datanode.DataNode: PacketResponder:
BP-1790420707-192.168.1.119-1353791420945:blk_8120590457789607396_342342, type=LAST_IN_PIPELINE,
downstreams=0:[] terminating
[...]
2012-12-14 06:56:37,076 ERROR org.apache.hadoop.hdfs.server.datanode.DataNode:
008920a.intranet.jgh:50010:DataXceiver error processing WRITE_BLOCK operation src: /
10.196.121.18:38695 dest: /10.196.121.18:50010
java.net.SocketTimeoutException: 60000 millis timeout while waiting for channel to be ready
for read. ch : java.nio.channels.SocketChannel[connected local=/10.196.121.18:50010 remote=/
10.196.121.18:38695]
    at org.apache.hadoop.net.SocketIOWithTimeout.doIO(SocketIOWithTimeout.java:165)
    at org.apache.hadoop.net.SocketInputStream.read(SocketInputStream.java:154)
    at org.apache.hadoop.net.SocketInputStream.read(SocketInputStream.java:127)
    at java.io.FilterInputStream.read(FilterInputStream.java:116)
    at java.io.BufferedInputStream.fill(BufferedInputStream.java:218)
    at java.io.BufferedInputStream.read1(BufferedInputStream.java:258)
    at java.io.BufferedInputStream.read(BufferedInputStream.java:317)
    at java.io.DataInputStream.read(DataInputStream.java:132)
    at org.apache.hadoop.io.IOUtils.readFully(IOUtils.java:192)
    at
org.apache.hadoop.hdfs.protocol.datatransfer.PacketReceiver.doReadFully(PacketReceiver.java:213)
    at
org.apache.hadoop.hdfs.protocol.datatransfer.PacketReceiver.doRead(PacketReceiver.java:134)
    at
org.apache.hadoop.hdfs.protocol.datatransfer.PacketReceiver.receiveNextPacket(PacketReceiver.java:1
09)
    at
org.apache.hadoop.hdfs.server.datanode.BlockReceiver.receivePacket(BlockReceiver.java:414)
    at
org.apache.hadoop.hdfs.server.datanode.BlockReceiver.receiveBlock(BlockReceiver.java:635)
    at org.apache.hadoop.hdfs.server.datanode.DataXceiver.writeBlock(DataXceiver.java:506)
    at org.apache.hadoop.hdfs.protocol.datatransfer.Receiver.opWriteBlock(Receiver.java:98)
    at org.apache.hadoop.hdfs.protocol.datatransfer.Receiver.processOp(Receiver.java:66)
    at org.apache.hadoop.hdfs.server.datanode.DataXceiver.run(DataXceiver.java:219)
    at java.lang.Thread.run(Thread.java:662)

```

### Diagnostic :

Le DataNode n'a pas pu soutenir la demande en écriture sur le disque, peut-être du à un manque de mémoire (swapping).

### Resolution :

1. Ajouter plus de DataNodes et ainsi alléger la charge des serveurs maitre Hadoop (NN,SNN,JT,Hive Server) ???

### References :

<https://groups.google.com/a/cloudera.org/forum/?fromgroups=#!topic/cdh-user/8kXX4rBy9U4>  
<http://blog.cloudera.com/blog/2009/03/configuration-parameters-what-can-you-just-ignore/>  
<https://issues.apache.org/jira/browse/HDFS-2906>

# Annexe G

## Connaissances en lien avec le bac de génie logiciel

Lors de ce PFE nous avons utilisé des notions provenant de 20 cours de génie logiciel. Voici les détails et justifications dans le tableau ci-bas :

TABLE G.1: Grille de relations entre ce PFE et les connaissances de génie logiciel

Sigle	Titre du Cours	Artefact / No- tion	Justification
COM110	Méthodes de communication	Compte-rendu de réunions	Communications avec le client

TABLE G.1: (suite)

Sigle	Titre du Cours	Artefact / No- tion	Justification
GIA400	Analyse de rentabilité de projets	Retour sur investissement (ROI)	Notre solution peut offrir un grand retour sur l'investissement, puisqu'elle utilise des logiciels gratuits et du matériel à faible coût.
GPO605	Entrepreneuriat et innovation	Innovation	Expérimentations avec les nouvelles technologies qui ne sont pas toujours adéquatement documentées, tout en respectant le cadre financier du client.
LOG120	Conception orientée objet	Design Patterns	Les vues d'une BD sont l'équivalent d'un patron façade.
LOG210	Analyse et conception de logiciels	Unified Process	Notre projet utilise plusieurs disciplines (analyse d'affaires, élicitations des exigences, analyse et conception, implémentation) répartis à travers plusieurs itérations et phases (inception, élaboration, construction).

TABLE G.1: (suite)

Sigle	Titre du Cours	Artefact / No- tion	Justification
LOG210	Analyse et conception de logiciels	Modèle du Do- maine	Présent dans la section 1.2 du document de Vision.
LOG240	Tests et main- tenance	Tests	Comparaison de deux sys- tèmes différents qui font les font les mêmes requêtes
LOG320	Structures de données et algorithmes	Hash, B-Tree, Compression	Utilisation de GZip, Snappy pour comparer les perfor- mances. Comparaison des performances des différents index d'Oracle.
LOG330	Assurance de la qualité des logi- ciels	ISO/IEC 9126	Exigences non-fonctionnelles présentes à la section 4.2 du document de Vision
LOG330	Assurance de la qualité des logi- ciels	ISO/IEC/IEEE 16085	Gestion des risques présentes dans le chapitre Inception du rapport final
LOG350	Interfaces utili- sateur	Loi de Fitz	Interface intuitive pour le Query Builder
LOG410	Analyses de be- soins et spécifi- cations	Vision	Voir le document de Vision

TABLE G.1: (suite)

Sigle	Titre du Cours	Artefact / No- tion	Justification
LOG410	Analyses de besoins et spécifications	IEEE 830	Voir le document de SRS
LOG430	Architecture logicielle	Diagramme des composantes	Présent dans le document d'Architecture.
LOG430	Architecture logicielle	Diagramme de déploiement	Présent dans le document d'Architecture.
LOG515	Gestion de projets en génie logiciel	Échéancier	Voir l'annexe sur le plan de travail
LOG610	Réseaux de télécommunication	TCP/IP, DNS	Formatter les noms d'hôtes et de port pour la communication client-serveur entre les différents services d'hadoop.

TABLE G.1: (suite)

Sigle	Titre du Cours	Artefact / No- tion	Justification
LOG619	Sécurité des systèmes	Ports ouverts, comptes utilisateurs	Dans la section “2.1 Composantes d’EndoMine” du document d’architecture, on démontre la répartition des données sécurisée, et on explique qu’un seul utilisateur y aura accès. A chaque niveau d’accès, il y’a un nouveau compte utilisateur. Et anonymisation.
LOG640	Introduction au traitement parallèle	Noeud de calcul	Configuration optimale du nombre de mappers et de reducers par noeud pour Map-Reduce

TABLE G.1: (suite)

Sigle	Titre du Cours	Artefact / No- tion	Justification
LOG660	Bases de données de haute performance	Algorithmes de jointure, fonctions avancées d'Oracle	Utilisé pour trouver la meilleure façon de fusionner des tables de plusieurs millions de rangées. Nous avons expérimenté avec des partitions, de la compression, des index globaux et locaux, hash, cluster index, etc. Analyse des plans d'exécution de Hive, Impala et Oracle.
LOG710	Principes des systèmes d'exploitation et programmation système	RAM, Swap	Nous avons effectués des recherches pour comprendre davantage comment fonctionne les systèmes d'exploitations pour arriver à prouver que le système "swappais".

TABLE G.1: (suite)

Sigle	Titre du Cours	Artefact / No- tion	Justification
LOG730	Introduction aux systèmes distribués	TCP/IP, DNS, système distribué	Bien qu'il n'y ait qu'une seule machine dans notre système, il s'agit d'un système distribué. Impossible donc d'éviter les problèmes réseaux (trop de connections simultanées). Serveur écoutant sur localhost seulement ou sur l'hôte complet. De plus il fallait gérer l'assignation du DNS de façon à pouvoir facilement porter la machine depuis la maison, à l'ÉTS et au JGH.
MAT210	Logique et mathématiques discrètes	Bitmap Index	Oracle peut faire des AND ou des OR et c'est pour cela qu'il est performant pour ce type d'index.
MAT350	Probabilités et statistiques	Statistiques	Récolter des statistiques sur l'utilisation des ressources existantes, et temps d'exécution.

# Annexe H

## Survol des produits Pentaho

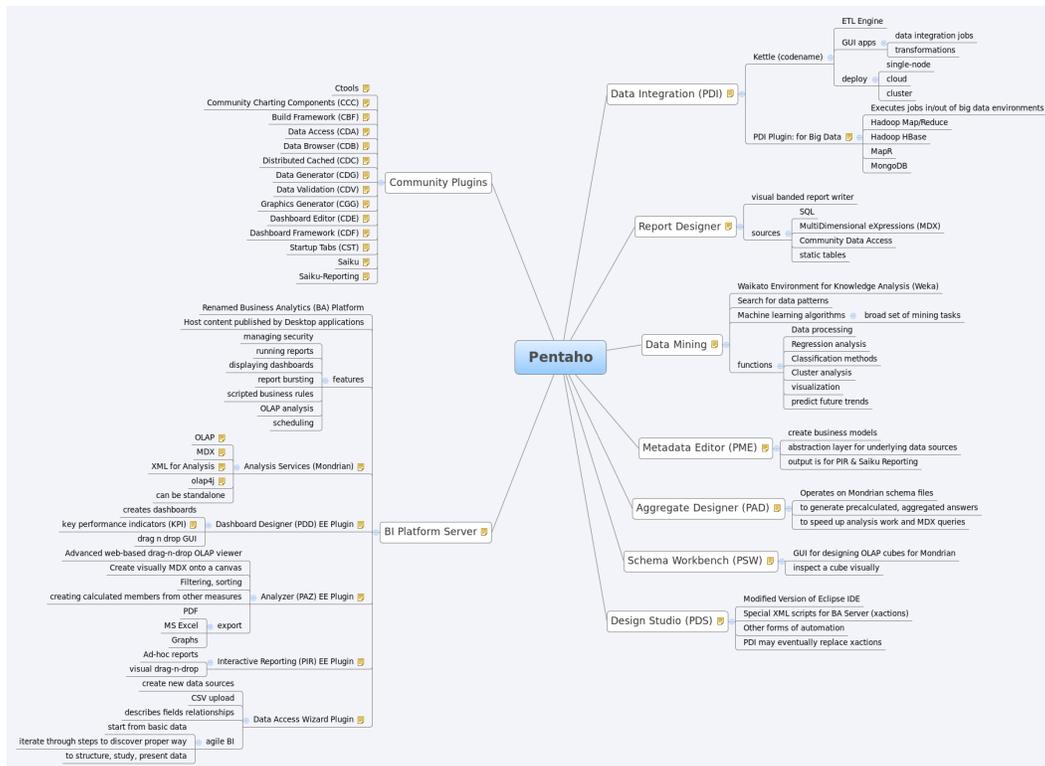


FIGURE H.1 – Mindmap des produits Pentaho

# Annexe I

## Installation de Hadoop

Listing I.1 – Installation de CDH4

```
1 # =====
2 # Oracle JAVA JDK Installation
3 # =====
4 cd Downloads/
5 chmod a+x jdk-6u34-linux-x64.bin
6 sudo ./jdk-6u34-linux-x64.bin
7 sudo mv jdk1.6.0_34/ /usr/lib/jvm/
8 ls -l /usr/lib/jvm
9 sudo ln -s /usr/lib/jvm/jdk1.6.0_34 /usr/lib/jvm/java-6-oracle
10 sudo update-alternatives --install /usr/bin/java java /usr/lib/jvm/java-6-
    oracle/jre/bin/java 2
11 sudo update-alternatives --install /usr/bin/javac javac /usr/lib/jvm/java-6-
    oracle/bin/javac 1
12 sudo update-alternatives --install /usr/bin/javaws javaws /usr/lib/jvm/java
    -6-oracle/bin/javaws 1
13 sudo update-alternatives --config java
14
15 # ----- Only for local testing -----
16 sudo update-alternatives --install /usr/lib/mozilla/plugins/mozilla-
    javaplugin.so mozilla-javaplugin.so /usr/lib/jvm/java-6-oracle/jre/lib/
    amd64/libnpjp2.so 1
17 sudo mkdir -p /opt/google/chrome/plugins
```

```
18 sudo update-alternatives --install /opt/google/chrome/plugins/chrome-
    javaplugin.so chrome-javaplugin.so /usr/lib/jvm/java-6-oracle/jre/lib/
    amd64/libnpp2.so 1
19 sudo update-alternatives --config java
20 sudo update-alternatives --config chrome-javaplugin.so
21 sudo update-alternatives --config mozilla-javaplugin.so
22 # —— Only for local testing end ——
23
24 sudo vi /etc/profile.d/java.sh
25     #!/bin/bash
26     export JAVA_HOME=$( dirname $( dirname $( dirname $( readlink -e /usr/bin
    /java ) ) ) )
27     export JDK_HOME=$JAVA_HOME
28     export JRE_HOME=$JAVA_HOME/jre
29     export PATH=$JAVA_HOME/bin:$PATH
30 sudo chmod +x /etc/profile.d/java.sh
31 source /etc/profile.d/java.sh
32 java -version
33
34
35 # =====
36 # Make sure the JAVA_HOME environment variable is set for the root user
37 # =====
38 sudo reboot
39 [CTRL]+[ALT]+[F1]
40 login as root
41 env | grep JAVA_HOME
42
43
44 # =====
45 # fix localhost problem
46 # =====
47 # Remove entries relating to 127.0.1.1 <COMPUTER NAME> from /etc/hosts
48 # Make sure /etc/hostname matches the value in /etc/hosts
49 $ cat /etc/hostname
50 localhost
51 $ cat /etc/hosts
52 127.0.0.1      localhost
53 # Note: Red Hat systems sets the hostname in /etc/sysconfig/network
54
```

```
55
56 # =====
57 # CDH4 Installation
58 # =====
59 # --- UBUNTU ---
60 # Pseudo distribue : https://ccp.cloudera.com/display/CDH4DOC/Installing+CDH4+on+a+Single+Linux+Node+in+Pseudo-distributed+Mode#InstallingCDH4on+a+Single+Linux+Node+in+Pseudo-distributed+Mode-InstallingCDH4withMRv1on+a+Single+Linux+Node+in+Pseudo+distributed+mode
61
62 sudo apt-get update
63
64 # The easiest way is to install the pseudo-distributed config, which will
65 # install all dependencies.
66 # Then, the config can be upgraded to a fully distributed cluster once
67 # more nodes are installed.
68 # Dependencies: hadoop hadoop-hdfs hadoop-hdfs-datanode hadoop-hdfs-
69 # namenode hadoop-hdfs-secondarynamenode
70 # hadoop-0.20-mapreduce hadoop-0.20-mapreduce-jobtracker
71 # hadoop-0.20-mapreduce-tasktracker
72 sudo apt-get install hadoop-0.20-conf-pseudo
73
74
75 # If error with JAVA_HOME not set run the following command to be a root
76 # add the following line into /etc/hadoop/conf/hadoop-env.sh
77 export JAVA_HOME=/usr/lib/jvm/jdk1.6.0_34
78
79 # Note: you WILL want to change the default data dir: /var/lib/hadoop-hdfs/
80 # cache/hdfs/dfs/data
81 sudo -u hdfs hdfs namenode -format
82 # Start HDFS services
83 sudo service hadoop-hdfs-namenode start
84 sudo service hadoop-hdfs-secondarynamenode start
85 sudo service hadoop-hdfs-datanode start
86 # Create Hadoop directory structure
87 sudo -u hdfs hadoop fs -mkdir /tmp
88 sudo -u hdfs hadoop fs -chmod -R 1777 /tmp
89 sudo -u hdfs hadoop fs -ls /
90 sudo -u hdfs hadoop fs -mkdir /var
91 sudo -u hdfs hadoop fs -mkdir /var/lib
```

```
87 sudo -u hdfs hadoop fs -mkdir /var/lib/hadoop-hdfs
88 sudo -u hdfs hadoop fs -mkdir /var/lib/hadoop-hdfs/cache
89 sudo -u hdfs hadoop fs -mkdir /var/lib/hadoop-hdfs/cache/mapred
90 sudo -u hdfs hadoop fs -mkdir /var/lib/hadoop-hdfs/cache/mapred/mapred
91 sudo -u hdfs hadoop fs -mkdir /var/lib/hadoop-hdfs/cache/mapred/mapred/
    staging
92 sudo -u hdfs hadoop fs -chmod 1777 /var/lib/hadoop-hdfs/cache/mapred/mapred/
    staging
93 sudo -u hdfs hadoop fs -chown -R mapred /var/lib/hadoop-hdfs/cache/mapred
94 sudo -u hdfs hadoop fs -ls -R /
95 # Start Map Reduce services
96 sudo service hadoop-0.20-mapreduce-jobtracker start
97 sudo service hadoop-0.20-mapreduce-tasktracker start
98
99
100 # =====
101 # Map Reduce Test
102 # =====
103 # Create a directory structure in HDFS for current user to run mapreduce jobs
104 sudo -u hdfs hadoop fs -mkdir /user/mysecretuser
105 sudo -u hdfs hadoop fs -chown mysecretuser /user/mysecretuser
106 sudo -u hdfs hadoop fs -ls -R /
107 # Grep all config that starts with dfs from the hadoop config
108 hadoop fs -mkdir input
109 hadoop fs -put /etc/hadoop/conf/*.xml input/
110 hadoop fs -ls input
111 hadoop jar /usr/lib/hadoop-0.20-mapreduce/hadoop-examples.jar grep input
    output 'dfs[a-z.]+'
112 hadoop fs -ls
113 hadoop fs -ls output
114 hadoop fs -cat output/part-00000 | head
```

## Listing I.2 – Installation de Hive

```
1
2 # https://ccp.cloudera.com/display/CDH4DOC/Hive+Installation#HiveInstallation
   -InstallingHive
3
4 # For the server, install Hive and Hive MetaStore
5 sudo apt-get install hive hive-metastore
6
7 # For datanodes, only the hive client should be required
8 sudo apt-get install hive
9
10 # Install the MySQL JDBC Connector in the Hive lib directory
11 curl -L 'http://www.mysql.com/get/Downloads/Connector-J/mysql-connector-java
   -5.1.22.tar.gz/from/http://mysql.he.net/' | tar xz
12 sudo cp mysql-connector-java-5.1.22/mysql-connector-java-5.1.22-bin.jar /usr/
   lib/hive/lib/
13
14 # Create the MySQL Database for the Hive MetaStore
15 mysql -uroot -p
16 mysql> CREATE DATABASE hivemetastore;
17 mysql> USE hivemetastore;
18 mysql> SOURCE /usr/lib/hive/scripts/metastore/upgrade/mysql/hive-schema
   -0.9.0.mysql.sql;
19 mysql> CREATE USER 'hive'@'localhost' IDENTIFIED BY 'your hive password for
   mysql';
20 mysql> CREATE USER 'hive'@'my.secret.host' IDENTIFIED BY 'your hive password
   for mysql';
21 mysql> GRANT ALL PRIVILEGES ON hivemetastore.* TO 'hive'@'localhost' WITH
   GRANT OPTION;
22 mysql> GRANT ALL PRIVILEGES ON hivemetastore.* TO 'hive'@'my.secret.host'
   WITH GRANT OPTION;
23 mysql> FLUSH PRIVILEGES;
24 mysql> quit;
25
26 # Configure Hive to find its MetaStore
27 sudo vi /etc/hive/conf/hive-site.xml
28     <property>
29         <name>javax.jdo.option.ConnectionURL</name>
30         <value>jdbc:mysql://localhost/hivemetastore</value>
31         <description>JDBC connect string for a JDBC metastore</description>
```

```
32     </property>
33
34     <property>
35         <name>javax.jdo.option.ConnectionDriverName</name>
36         <value>com.mysql.jdbc.Driver</value>
37         <description>Driver class name for a JDBC metastore</description>
38     </property>
39
40     <property>
41         <name>javax.jdo.option.ConnectionUserName</name>
42         <value>hive</value>
43     </property>
44
45     <property>
46         <name>javax.jdo.option.ConnectionPassword</name>
47         <value> your hive password for mysql </value>
48     </property>
49
50     <property>
51         <name>datanucleus.autoCreateSchema</name>
52         <value>false</value>
53     </property>
54
55     <property>
56         <name>datanucleus.fixedDatastore</name>
57         <value>true</value>
58     </property>
59
60 # Starts MetaStore
61 sudo service hive-metastore start
62
63 # Further steps required to setup HiveServer2
64 # To avoid potential incompatibility , we sticked with Hive Server 1
65
66 # [edit] We eventually also installed a standalone hive-server
67 # The reason to install this package , is that it does not require the
68   administrator
69 # to share JDBC login information for the Metastore database with each Hive
70   user .
71 sudo apt-get install hive-server
```

```
70
71
72 # Create hive directories and gives ownership to whichever user will run hive
73 # (here we used 'mysecretuser', but perhaps 'hive' user is more appropriate)
74 sudo -u hdfs hadoop fs -mkdir /user/hive/warehouse
75 sudo -u hdfs hadoop fs -chmod g+w /user/hive/warehouse
76 sudo -u hdfs hadoop fs -chown mysecretuser:mysecretuser /user/hive/warehouse
77
78
79 # =====
80 # Testing Hive Installation (as mysecretuser user)
81 # =====
82 hive
83 hive> CREATE TABLE pokes (foo INT, bar STRING);
84 hive> SHOW TABLES;
85 hive> CREATE TABLE invites (foo INT, bar STRING) PARTITIONED BY (ds STRING);
86 hive> SHOW TABLES '.*s';
87 hive> DESCRIBE invites;
88 hive> ALTER TABLE pokes ADD COLUMNS (new_col INT);
89 hive> ALTER TABLE invites ADD COLUMNS (new_col2 INT COMMENT 'a comment');
90 hive> ALTER TABLE invites RENAME TO events;
91 hive> SHOW TABLES;
92 hive> DROP TABLE events;
93 hive> DROP TABLE POKES;
94 hive> exit;
95
96 # =====
97 # Using Hive and HBase
98 # =====
99 # Add these lines at the top of every hive scripts:
100 ADD JAR /usr/lib/hive/lib/zookeeper.jar;
101 ADD JAR /usr/lib/hive/lib/hbase.jar;
102 ADD JAR /usr/lib/hive/lib/hive-hbase-handler-0.9.0-cdh4.1.0.jar
103 ADD JAR /usr/lib/hive/lib/guava-r09.jar;
104
105 # =====
106 # Much more examples
107 # =====
108 # https://cwiki.apache.org/confluence/display/Hive/GettingStarted
```

## Listing I.3 – Installation de Impala

```

1 # https://ccp.cloudera.com/display/IMPALA10BETADOC/Installing+Impala
2 # https://ccp.cloudera.com/display/IMPALA10BETADOC/Using+Impala
3 # https://ccp.cloudera.com/display/IMPALA10BETADOC/Configuring+Impala+for+
   Performance+without+Cloudera+Manager
4
5 # =====
6 # = BETA REQUIREMENTS
7 # =====
8 # * RHEL/CentOS 6.2 systems
9 # * Hive with external DB (NOT derby)
10
11
12 # =====
13 # = INSTALL
14 # =====
15 # As root:
16 cd /etc/yum.repos.d/
17 wget http://beta.cloudera.com/impala/redhat/6/x86_64/impala/cloudera-impala.
   repo
18
19 yum install impala
20 yum install impala-shell
21
22 mkdir -p /usr/lib/impala/conf
23 ln -s /etc/hadoop/conf/core-site.xml /usr/lib/impala/conf/
24 ln -s /etc/hadoop/conf/hdfs-site.xml /usr/lib/impala/conf/
25 ln -s /etc/hive/conf/hive-site.xml /usr/lib/impala/conf/
26 ln -s /etc/hadoop/conf/log4j.properties /usr/lib/impala/conf/
27 ls -l /usr/lib/impala/conf/
28
29 # More steps required to enable Kerberos authentication
30
31
32
33 # =====
34 # = CONFIGURATION
35 # =====
36
37 vi /etc/hadoop/conf/core-site.xml

```

```
38 <!-- IMPALA REQUIREMENTS (start) -->
39 <property>
40   <name>dfs.client.read.shortcircuit</name>
41   <value>true</value>
42 </property>
43
44 <property>
45   <name>dfs.client.read.shortcircuit.skip.checksum</name>
46   <value>>false</value>
47 </property>
48 <!-- IMPALA REQUIREMENTS (end) -->
49
50 vi /etc/hadoop/conf/hdfs-site.xml
51 <!-- IMPALA REQUIREMENTS (start) -->
52 <!-- Permissions for the directories where data node stores its blocks.
53   -->
54 <property>
55   <name>dfs.datanode.data.dir.perm</name>
56   <value>750</value>
57 </property>
58
59 <!-- Enable Impala to call getBlockLocalPathInfo -->
60 <!-- [David] It seems that all users running mapreduce jobs must also be in
61   this list. -->
62 <property>
63   <name>dfs.block.local-path-access.user</name>
64   <value>impala,hdfs,mapred,mysecretuser</value>
65 </property>
66
67 <!-- Enables data locality support for the DistributedFileSystem#
68   getFileVBlockStorageLocations API. -->
69 <property>
70   <name>dfs.datanode.hdfs-blocks-metadata.enabled</name>
71   <value>true</value>
72 </property>
73 <!-- IMPALA REQUIREMENTS (end) -->
74
75 # Add all users in dfs.block.local-path-access.user to the HDFS group
76 # All these users may not be required, but this works for me
77 usermod -a -G hdfs impala
```

```
75 usermod -a -G hdfs mapred
76 usermod -a -G hdfs mysecretuser
77 # If required, to remove from a group: gpasswd -d mapred hdfs
78
79
80 # [EDIT] This makes sure the impala user is a member of these groups (and
      that impala is the default group)
81 usermod -g impala -a -G hdfs,myscretuser impala
82
83 # IMPORTANT: here's the permissions that ended working for us:
84     uid=494(hdfs) gid=487(hdfs) groups=487(hdfs),488(hadoop)
85     uid=491(mapred) gid=484(mapred) groups=484(mapred),487(hdfs)
86     uid=484(impala) gid=477(impala) groups=477(impala),500(myscretuser)
      ,487(hdfs)
87     uid=500(myscretuser) gid=500(myscretuser) groups=500(myscretuser)
      ,487(hdfs)
88 # To verify your current permissions, run this command:
89 id hdfs; id mapred; id impala; id mysecretuser
90
91
92 # On data directory, change permission to match dfs.datanode.data.dir.perm
93 chmod -R 750 /data{1,2,3}/dfs/dn
94
95 # Restart Impala and HDFS (and also mapreduce if you've added it to the hdfs
      group)
96 service hadoop-hdfs-secondarynamenode stop
97 service hadoop-hdfs-datanode stop
98 service hadoop-hdfs-namenode stop
99 service hadoop-hdfs-datanode start
100 service hadoop-hdfs-namenode start
101 service hadoop-hdfs-namenode status
102 service hadoop-hdfs-datanode status
103 service hadoop-hdfs-secondarynamenode start
104 service hadoop-hdfs-secondarynamenode status
105
106
107 # =====
108 # = STARTING IMPALA MANUALLY
109 # =====
110
```

```
111 # Start Impala State Store (GLOG_v=1 increases logging)
112 GLOG_v=1 impala-state-store -state_store_port=24000
113
114 # Start Impala Daemon on each data node
115 MYIP='ifconfig eth0 | grep "inet_addr" | awk -F: '{print $2}' | awk '{print
    $1}' '
116 GLOG_v=1 impalad -state_store_host=my.secret.host -nn=my.secret.host -nn_port
    =8020 hostname=my.secret.host -ipaddress=$MYIP
117
118 # To enable more verbose logging, uses GLOG_v=2
119
120 # Checking Impala Configuration Values
121 http://<hostname>:<port>/varz
122 http://localhost:25000/varz
123
124
125 # =====
126 # = SETTING UP IMPALA initd scripts
127 # =====
128 [root@localhost ~]# mv impalad /etc/init.d/impalad
129 [root@localhost ~]# mv impala-state-store /etc/init.d/impala-state-store
130 [root@localhost ~]# chmod 755 /etc/init.d/impalad /etc/init.d/impala-state-
    store
131 [root@localhost ~]# ls -l /etc/init.d/impala*
132 -rwxr-xr-x 1 root root 1637 Dec 13 23:15 /etc/init.d/impalad
133 -rwxr-xr-x 1 root root 1597 Dec 13 23:21 /etc/init.d/impala-state-store
134 [root@localhost ~]# chkconfig --add impalad
135 [root@localhost ~]# chkconfig --add impala-state-store
136 [root@localhost ~]# chkconfig impala-state-store on
137 [root@localhost ~]# chkconfig impalad on
138 [root@localhost ~]# chkconfig |grep impala
```

Listing I.4 – Notre propre script init.d pour impala-state-store

```

1  #!/bin/sh
2  #
3  # Startup script for the Impala State Store
4  #
5  # chkconfig: - 85 15
6  # description: Hadoop Impala State Store
7  # processname: impala-state-store
8  # pidfile: /var/run/impala-state-store.pid
9
10 # Set environment.
11 RUNUSER=impala
12 SERVICE_NAME="impala-state-store"
13 LOCKFILE="/var/lock/$SERVICE_NAME"
14
15
16
17 # Source function library.
18 . /etc/rc.d/init.d/functions
19
20
21 # See how we were called.
22 case "$1" in
23     start)
24         # Start daemon.
25         echo -n "Starting_$SERVICE_NAME:_"
26
27         MYHOST='hostname '
28         export GLOG_v=2
29         export GLOG_log_dir=/var/log/impala/
30
31         # This alternative also works:
32         #su $RUNUSER -p -c "impala-state-store -log_filename='impala-state-
33             store' -state_store_host='$MYHOST' -state_store_port=24000 &> /dev
34             /null &"
35
36         daemon --user $RUNUSER "impala-state-store_-log_filename='impala-
37             state-store'_-state_store_host='$MYHOST'_-state_store_port=24000_
38             &>_/dev/null_&"

```

```
36     RETVAL=$?
37     echo
38     [ $RETVAL = 0 ] && touch $LOCKFILE
39     ;;
40 stop)
41     # Stop daemons.
42     echo -n "Shutting down $SERVICE_NAME: "
43     killproc $SERVICE_NAME
44     RETVAL=$?
45     echo
46     [ $RETVAL = 0 ] && rm -f $LOCKFILE
47     ;;
48 restart)
49     $0 stop
50     $0 start
51     ;;
52 status)
53     PID=$(pidof $SERVICE_NAME) || true
54     if [ -n "$PID" ]; then
55         echo "$SERVICE_NAME is running (pid $PID)."
56         exit 0
57     else
58         echo "$SERVICE_NAME is NOT running."
59         if [ -e "$PIDFILE" ]; then
60             exit 1
61         else
62             exit 3
63         fi
64     fi
65     ;;
66 *)
67     echo "Usage: $0 {start | stop | restart | status}"
68     exit 1
69 esac
70
71 exit 0
```

Listing I.5 – Notre propre script init.d pour impalad

```

1  #!/bin/sh
2  #
3  # Startup script for the Impala Daemon
4  #
5  # chkconfig: - 85 15
6  # description: Hadoop Impala Daemon
7  # processname: impalad
8  # pidfile: /var/run/impalad.pid
9
10 # Set environment.
11 RUNUSER=impala
12 SERVICE_NAME="impalad"
13 LOCKFILE="/var/lock/$SERVICE_NAME"
14
15 # Configure these to match your Namenode and Impala StateStore configurations
16 NN_HOST="my.secret.host"
17 NN_PORT=8020
18 SS_HOST="my.secret.host"
19
20
21 # Source function library.
22 . /etc/rc.d/init.d/functions
23
24
25 # See how we were called.
26 case "$1" in
27     start)
28         # Start daemon.
29         echo -n "Starting _$SERVICE_NAME:_"
30
31         MYIP='ifconfig eth0 | grep "inet_addr" | awk -F: '{print $2}' | awk
32             '{print $1}'
33         MYHOST='hostname '
34         export GLOG_v=2
35         export GLOG_log_dir=/var/log/impalad/
36         daemon --user $RUNUSER "impalad_-nn='$NN_HOST' _-nn_port=$NN_PORT_-
37             state_store_host='$SS_HOST' _hostname='$MYHOST' _ipaddress=$MYIP_-
38             log_filename='impalad' _&>_/dev/null_&"
39         #sleep 1

```

```
37     RETVAL=$?
38     echo
39     [ $RETVAL = 0 ] && touch $LOCKFILE
40     ;;
41 stop)
42     # Stop daemons.
43     echo -n "Shutting down $SERVICE_NAME:_"
44     killproc $SERVICE_NAME
45     RETVAL=$?
46     echo
47     [ $RETVAL = 0 ] && rm -f $LOCKFILE
48     ;;
49 restart)
50     $0 stop
51     $0 start
52     ;;
53 status)
54     PID=$(pidof $SERVICE_NAME) || true
55     if [ -n "$PID" ]; then
56         echo "$SERVICE_NAME is running (pid $PID)."
57         exit 0
58     else
59         echo "$SERVICE_NAME is NOT running."
60         if [ -e "$PIDFILE" ]; then
61             exit 1
62         else
63             exit 3
64         fi
65     fi
66     ;;
67 *)
68     echo "Usage: $0 {start|stop|restart|status}"
69     exit 1
70 esac
71
72 exit 0
```

# Annexe J

## Configuration de Hadoop

Listing J.1 – core-site.xml

```
1 <?xml version="1.0"?>
2 <?xml-stylesheet type="text/xsl" href="configuration.xsl"?>
3
4 <configuration>
5   <property>
6     <name>fs.default.name</name>
7     <value>hdfs://my.secret.host:8020</value>
8   </property>
9
10  <!-- OOZIE proxy user setting -->
11  <property>
12    <name>hadoop.proxyuser.oozie.hosts</name>
13    <value>*</value>
14  </property>
15  <property>
16    <name>hadoop.proxyuser.oozie.groups</name>
17    <value>*</value>
18  </property>
19
20  <!-- HTTPFS proxy user setting -->
21  <property>
22    <name>hadoop.proxyuser.httpfs.hosts</name>
23    <value>*</value>
```

```
24 </property>
25 <property>
26   <name>hadoop.proxyuser.httpfs.groups</name>
27   <value>*</value>
28 </property>
29
30 <!-- The default block size for new files. (DEFAULT = 67108864 = 64 MB) -->
31 <property>
32   <name>dfs.block.size</name>
33   <value>268435456</value>   <!-- 256 MB -->
34 </property>
35
36 <!-- The size of buffer for use in sequence files. The size of this buffer
      should probably be a multiple of hardware page size (4096 on Intel x86),
      and it determines how much data is buffered during read and write
      operations (DEFAULT = 4096) -->
37 <property>
38   <name>io.file.buffer.size</name>
39   <value>65536</value>
40 </property>
41
42 <!-- Comma-separated list of the compression codec classes that can be used
      for compression/decompression (DEFAULT="") -->
43 <property>
44   <name>io.compression.codecs</name>
45   <value>org.apache.hadoop.io.compress.DefaultCodec,org.apache.hadoop.io.
      compress.GzipCodec,org.apache.hadoop.io.compress.SnappyCodec</value>
46 </property>
47
48
49 <!-- IMPALA REQUIREMENTS (start) -->
50 <property>
51   <name>dfs.client.read.shortcircuit</name>
52   <value>>true</value>
53 </property>
54
55 <property>
56   <name>dfs.client.read.shortcircuit.skip.checksum</name>
57   <value>>false</value>
58 </property>
```

```
59 <!-- IMPALA REQUIREMENTS (end) -->
60
61 </configuration>
```

Listing J.2 – hdfs-site.xml

```

1 <?xml version="1.0"?>
2 <?xml-stylesheet type="text/xsl" href="configuration.xsl"?>
3
4 <configuration>
5   <property>
6     <name>dfs.replication</name>
7     <value>1</value>
8   </property>
9   <!-- Immediately exit safemode as soon as one DataNode checks in.
10      On a multi-node cluster, these configurations must be removed. -->
11   <property>
12     <name>dfs.safemode.extension</name>
13     <value>0</value>
14   </property>
15   <property>
16     <name>dfs.safemode.min.datanodes</name>
17     <value>1</value>
18   </property>
19
20   <property>
21     <name>dfs.namenode.name.dir</name>
22     <value>/var/lib/hadoop-hdfs/cache/${user.name}/dfs/name</value>
23   </property>
24   <property>
25     <name>dfs.namenode.checkpoint.dir</name>
26     <value>/var/lib/hadoop-hdfs/cache/${user.name}/dfs/namesecondary</value>
27   </property>
28
29   <!-- A base for other temporary directories (DEFAULT = /var/lib/hadoop-hdfs
30      /cache/${user.name}).
31      (changing to /data3/tmp/${user.name} didn't work.) -->
32   <property>
33     <name>hadoop.tmp.dir</name>
34     <value>/var/lib/hadoop-hdfs/cache/${user.name}</value>
35   </property>
36   <!-- Path(s) on the local filesystem where data node stores its blocks. (
37      default=/var/lib/hadoop-hdfs/cache/${user.name}/dfs/data) -->

```

```
38 <<<<<name>dfs.datanode.data.dir</name>
39 <<<<<value>/data1/dfs/dn,/data2/dfs/dn,/data3/dfs/dn</value>
40 <<<</property>
41
42 <<<<!-- Reserved space in bytes per volume. Always leave this much space free
      for non-dfs use (DEFAULT=0) -->
43 <<<<property>
44 <<<<<name>dfs.datanode.du.reserved</name>
45 <<<<<value>268435456</value><!-- 256 GB -->
46 <<<</property>
47
48
49 <<<<!-- IMPALA_REQUIREMENTS ( start ) -->
50 <<<<!-- Permissions for the directories where data node stores its blocks -->
      -->
51 <<<<property>
52 <<<<<name>dfs.datanode.data.dir.perm</name>
53 <<<<<value>750</value>
54 <<<</property>
55
56 <<<<!-- Enable Impala to call getBlockLocalPathInfo -->
57 <<<<!-- [ David ] It seems that all user running mapreduce jobs must be in this
      list -->
58 <<<<property>
59 <<<<<name>dfs.block.local-path-access.user</name>
60 <<<<<value>impala,hdfs,mapred,mysecretuser</value>
61 <<<<<!--value>impala</value -->
62 <<<</property>
63
64 <<<<!-- Enables data locality support for the DistributedFileSystem#
      getFileVBlockStorageLocations API -->
65 <<<<property>
66 <<<<<name>dfs.datanode.hdfs-blocks-metadata.enabled</name>
67 <<<<<value>>true</value>
68 <<<</property>
69 <<<<!-- IMPALA_REQUIREMENTS ( end ) -->
70
71 </configuration>
```

## Listing J.3 – mapred-site.xml

```

1 <?xml version="1.0"?>
2 <?xml-stylesheet type="text/xsl" href="configuration.xsl"?>
3
4 <configuration>
5   <property>
6     <name>mapred.job.tracker</name>
7     <value>http://my.secret.host:8021</value>
8   </property>
9
10  <!-- The local directory where MapReduce stores intermediate data files.
11       May be a comma-separated list of directories on different devices in
12       order to spread disk i/o. Directories that do not exist are ignored (
13       DEFAULT = ${hadoop.tmp.dir}/mapred/local). -->
14  <property>
15     <name>mapred.local.dir</name>
16     <value>/data1/tmp/mapred/local,/data2/tmp/mapred/local,/data3/tmp/mapred/
17         local</value>
18   </property>
19
20  <!-- The directory where MapReduce stores control files. -->
21  <property>
22     <name>mapred.system.dir</name>
23     <value>${hadoop.tmp.dir}/mapred/system</value>
24   </property>
25
26  <!-- Java opts for the task tracker child processes. (DEFAULT = -Xmx200m)
27       -->
28  <property>
29     <name>mapred.child.java.opts</name>
30     <!-- Starts JVM with <Xms> MB, but can increase up to <Xmx> MB -->
31     <value>-Xms3072m -Xmx3072m -XX:+HeapDumpOnOutOfMemoryError -
32         XX:HeapDumpPath=/var/log/hadoop-0.20-mapreduce/ </value>
33   </property>
34
35  <!-- The number of milliseconds before a task will be terminated if it
36       neither reads an input, writes an output, nor updates its status string.
37       -->
38  <!-- 900000ms = 900s = 15min (DEFAULT = 600000ms) -->
39  <property>

```

```
32 <name>mapred.task.timeout</name>
33 <value>900000</value>
34 </property>
35
36 <!-- The maximum number of map tasks that will be run simultaneously by a
    task tracker. (DEFAULT = 2) -->
37 <property>
38 <name>mapred.tasktracker.map.tasks.maximum</name>
39 <value>3</value>
40 <final>true</final>
41 </property>
42
43 <!-- The maximum number of reduce tasks that will be run simultaneously by
    a task tracker (DEFAULT = 2) -->
44 <property>
45 <name>mapred.tasktracker.reduce.tasks.maximum</name>
46 <value>3</value>
47 <final>true</final>
48 </property>
49
50 <!-- Number of streams to merge at once while sorting files. This
    determines the number of open file handles. (DEFAULT = 10) -->
51 <property>
52 <name>io.sort.factor</name>
53 <value>16</value>
54 </property>
55
56 <!-- Total amount of buffer memory (MB) to use while sorting files. (
    DEFAULT = 100) -->
57 <property>
58 <name>io.sort.mb</name>
59 <value>256</value>
60 </property>
61
62 <!-- Should the outputs of the maps be compressed (SequenceFile) before
    being sent across the network (DEFAULT = false) -->
63 <property>
64 <name>mapred.compress.map.output</name>
65 <value>true</value>
66 </property>
```

```
67
68 <!-- How should outputs of map be compressed? NONE, RECORD or BLOCK (
        DEFAULT = RECORD) -->
69 <property>
70   <name>mapred.output.compression.type</name>
71   <value>BLOCK</value>
72 </property>
73
74 <!-- Map Output Compression Code (DEFAULT = org.apache.hadoop.io.compress.
        DefaultCodec) -->
75 <property>
76   <name>mapred.map.output.compression.codec</name>
77   <value>org.apache.hadoop.io.compress.SnappyCodec</value>
78 </property>
79
80 <!-- How many tasks will re-use the same jvm (DEFAULT = 1) -->
81 <property>
82   <name>mapred.job.reuse.jvm.num.tasks</name>
83   <value>-1</value>
84 </property>
85
86 <!-- If true, then multiple instances of some map tasks may be executed in
        parallel. (DEFAULT = true) -->
87 <property>
88   <name>mapred.map.tasks.speculative.execution</name>
89   <value>>false</value>
90 </property>
91
92 <!-- If true, then multiple instances of some reduce tasks may be executed
        in parallel. (DEFAULT = true) -->
93 <property>
94   <name>mapred.reduce.tasks.speculative.execution</name>
95   <value>>false</value>
96 </property>
97
98 <!-- Fraction of the number of maps in the job which should be complete
        before reduces are scheduled for the job. (DEFAULT = 0.05) -->
99 <property>
100  <name>mapred.reduce.slowstart.completed.maps</name>
```

```
101     <value>0.99</value> <!-- aka do not start reducers until mappers are
      complete -->
102 </property>
103
104 </configuration>
```

## Listing J.4 – hive-site.xml

```

1 <?xml version="1.0"?>
2 <!--
3   Licensed to the Apache Software Foundation (ASF) under one or more
4   contributor license agreements. See the NOTICE file distributed with
5   this work for additional information regarding copyright ownership.
6   The ASF licenses this file to You under the Apache License, Version 2.0
7   (the "License"); you may not use this file except in compliance with
8   the License. You may obtain a copy of the License at
9
10   http://www.apache.org/licenses/LICENSE-2.0
11
12   Unless required by applicable law or agreed to in writing, software
13   distributed under the License is distributed on an "AS-IS" BASIS,
14   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
15   See the License for the specific language governing permissions and
16   limitations under the License.
17   -->
18 <?xml-stylesheet type="text/xsl" href="configuration.xsl"?>
19
20 <configuration>
21
22 <!-- Hive Configuration can either be stored in this file or in the hadoop
23   configuration files -->
24 <!-- that are implied by Hadoop setup variables.
25
26   -->
27
28 <!-- Aside from Hadoop setup variables - this file is provided as a
29   convenience so that Hive -->
30 <!-- users do not have to edit hadoop configuration files (that may be
31   managed as a centralized -->
32 <!-- resource).
33
34   -->
35
36 <!-- Hive Execution Parameters -->
37
38 <property>
39   <name>javax.jdo.option.ConnectionURL</name>
40   <value>jdbc:mysql://localhost/hivemetastore</value>
41   <description>JDBC connect string for a JDBC metastore</description>

```

```
34 </property>
35
36 <property>
37   <name>javax.jdo.option.ConnectionDriverName</name>
38   <value>com.mysql.jdbc.Driver</value>
39   <description>Driver class name for a JDBC metastore</description>
40 </property>
41
42 <property>
43   <name>javax.jdo.option.ConnectionUserName</name>
44   <value>hive</value>
45 </property>
46
47 <property>
48   <name>javax.jdo.option.ConnectionPassword</name>
49   <value>hive</value>
50 </property>
51
52 <property>
53   <name>datanucleus.autoCreateSchema</name>
54   <value>>false</value>
55 </property>
56
57 <property>
58   <name>datanucleus.fixedDatastore</name>
59   <value>>true</value>
60 </property>
61
62
63 <!-- Hive Statistics : Storage (DEFAULT = "jdbc:derby:;databaseName=
        TempStatsStore;create=true") -->
64 <!--
65 I don't know who coded this, but it's impossible to set a mysql password
        either using ';' or using '&'.
66 Creating a hivestats user at least stop generating errors on INSERT OVERWRITE
        .
67
68 Best solution would probably be to use a Derby solution using a complete path
        .
69 But it may be required to call COMPUTE STATISTICS on every nodes.
```

```
70 Check:
71 http://www.hiregion.com/2010/01/hive-metastore-derby-db.html
72 http://blog.cloudera.com/blog/2012/08/column-statistics-in-hive/
73 http://mail-archives.apache.org/mod_mbox/hive-user/201103.mbox/%3C4D7279B0
   .2070905@gatech.edu%3E
74 http://mail-archives.apache.org/mod_mbox/hive-user/201108.mbox/%3
   CCABezZmjAddMSv08aBSeyfNXbbv=6WzRr8hJ0q7=VBv+aA+Uxmg@mail.gmail.com%3E
75 —>
76 <property>
77   <name>hive.stats.dbconnectionstring</name>
78   <value>jdbc:mysql://localhost/hivestatistics?user=hivestats</value>
79 </property>
80
81 <!-- Hive Statistics : Driver (DEFAULT = "org.apache.derby.jdbc.
   EmbeddedDriver") —>
82 <property>
83   <name>hive.stats.jdbcdriver</name>
84   <value>com.mysql.jdbc.Driver</value>
85 </property>
86
87 </configuration>
```

## Listing J.5 – hive-env.sh

```
1 # The heap size of the jvm started by hive shell script can be controlled via :
2 #
3 export HADOOP_HEAPSIZE=1024
4 #
5 # Larger heap size may be required when running queries over large number of
6 #   files or partitions .
7 # By default hive shell scripts use a heap size of 256 (MB). Larger heap
8 #   size would also be
9 # appropriate for hive server (hwi etc).
```

# Annexe K

## Code SQL pour Oracle

Listing K.1 – Création des tables anonymisées

```
1  -- Creating Anonymous patient
2  -- These scripts must be executed as the owner of the staging db.
3  DROP TABLE V_P_LAB_PATIENT_ANON;
4  /
5  DROP TABLE V_P_LAB_PATIENT_LINK;
6  /
7  DROP TRIGGER V_P_LAB_PATIENT_LINK_INSERT;
8  /
9  DROP SEQUENCE SEQ_V_P_PATIENT_LINK_ANON_ID;
10 /
11 CREATE TABLE V_P_LAB_PATIENT_ANON
12 (
13     ANON_ID NUMBER(14,0) ,
14     DATE_OF_BIRTH NUMBER(10,0) ,
15     SEX CHAR(1)
16 );
17
18 CREATE UNIQUE INDEX PK_V_P_LAB_PATIENT_ANON ON V_P_LAB_PATIENT_ANON (ANON_ID)
19     ;
20 ALTER TABLE V_P_LAB_PATIENT_ANON ADD CONSTRAINT PK_V_P_LAB_PATIENT_ANON
21     PRIMARY KEY (ANON_ID) ;
```

```
22  —Creating Anonymous link table
23  CREATE TABLE V_P_LAB_PATIENT_LINK
24  (
25    PATIENT_AA_ID NUMBER(14,0) ,
26    ANON_ID NUMBER(14,0)
27  );
28  /
29  CREATE UNIQUE INDEX PK_V_P_LAB_PATIENT_LINK ON V_P_LAB_PATIENT_LINK (
30    PATIENT_AA_ID);
31  ALTER TABLE V_P_LAB_PATIENT_LINK ADD CONSTRAINT PK_V_P_LAB_PATIENT_LINK
32    PRIMARY KEY (PATIENT_AA_ID);
33  /
34  — Create sequence for V_P_PATIENT_LINK
35  /
36  CREATE SEQUENCE SEQ_V_P_PATIENT_LINK_ANON_ID START WITH 1 INCREMENT BY 1;
37  /
38  — Creating a more advanced trigger.
39  create or replace
40  TRIGGER V_P_LAB_PATIENT_LINK_INSERT
41  BEFORE INSERT ON V_P_LAB_PATIENT_LINK FOR EACH ROW
42  — *****
43  — CODE HIDDEN FOR SECURITY PURPOSES
44  — *****
45  END;
46  /
47  —
```

## Listing K.2 – Anonymisation des données sensibles

```

1  --FILL LINK
2  INSERT INTO V_P_LAB_PATIENT_LINK (PATIENT_AA_ID)
3  SELECT AA_ID FROM V_P_LAB_PATIENT;
4  /
5  CREATE INDEX IDX_V_P_LAB_PATIENT_LINK_AN_ID ON V_P_LAB_PATIENT_LINK (ANON_ID)
6  ;
7  /
8  INSERT INTO V_P_LAB_PATIENT_ANON (ANON_ID,DATE_OF_BIRTH,SEX)
9  SELECT l.ANON_ID, p.DATE_OF_BIRTH, p.SEX FROM V_P_LAB_PATIENT_LINK l ,
10         V_P_LAB_PATIENT p
11  WHERE l.PATIENT_AA_ID = p.AA_ID;
12  /
13  DROP INDEX CREATE INDEX IDX_PA_ST_AN_PA_ANON_AA_ID;
14  /
15  DROP TABLE V_P_LAB_PAT_STAY_ANON;
16  /
17  CREATE TABLE V_P_LAB_PAT_STAY_ANON
18  AS
19  SELECT s.AA_ID, s.CLINIC_ID, s.ROOM, s.BED, s.DOCTOR_ID, s.ADMISSION_DATE, s .
20         DISCHARGE_DATE, s.DIAGNOSIS1_ID, s.DIAGNOSIS2_ID, s.DIAGNOSIS3_ID, s .
21         ACCIDENT_CODE, s.TOTAL_CHARGES, s.TOTAL_PAYMENTS, s.PATIENT_BAL, s.ADMIT_FLAG,
22         s.DISCHARGE_FLAG, s.ADMIT_OUTP_FLAG, s.RESULT_CHANGED_FLAG, s .
23         DISCH_REP_PRINTED, s.JUST_POSTED_HIS_FLAG, s.DELETED_FLAG, s.ADMISSION_TIME, s .
24         .ADMISSION_DT, s.DISCHARGE_TIME, s.DISCHARGE_DT, s.ADMITTING_DOCTOR_ID, s .
25         CONSULTING_DOCTOR_ID, s.MEDICAL_SERVICE_ID, s.INSURANCE1_ID, s.INSURANCE2_ID,
26         s.INSURANCE3_ID, s.ACTIVE_FLAG, l.ANON_ID as PATIENT_ANON_ID,
27  p.ANON_ID as P_ANON_ID,
28  p.DATE_OF_BIRTH as P_DATE_OF_BIRTH,
29  p.SEX as P_SEX
30  FROM V_P_LAB_STAY s, V_P_LAB_PATIENT_LINK l, V_P_LAB_PATIENT_ANON p
31  WHERE s.PATIENT_AA_ID = l.PATIENT_AA_ID and p.ANON_ID = l.ANON_ID; --123s
32  /
33  CREATE UNIQUE INDEX PK_V_P_LAB_PAT_STAY_ANON ON V_P_LAB_PAT_STAY_ANON (AA_ID)
34  ; --15s
35  ALTER TABLE V_P_LAB_PAT_STAY_ANON ADD CONSTRAINT PK_V_P_LAB_PAT_STAY_ANON
36         PRIMARY KEY (AA_ID);
37  /
38  CREATE INDEX IDX_PA_ST_AN_PA_ANON_AA_ID ON V_P_LAB_PAT_STAY_ANON(
39         PATIENT_ANON_ID); --15s

```

# Annexe L

## License BSD

Tout le code, schémas, analyse, conceptions ou tout autre artéfact produit lors de ce projet est livré avec license BSD ci-bas.

### Listing L.1 – License BSD

```
1 Copyright (c) 2012, Anton Zakharov, David Lauzon, ETS
2 All rights reserved.
3
4 Redistribution and use in source and binary forms, with or without
5 modification, are permitted provided that the following conditions are met:
6
7 1. Redistributions of source code must retain the above copyright notice,
8   this
9   list of conditions and the following disclaimer.
10 2. Redistributions in binary form must reproduce the above copyright notice,
11   this list of conditions and the following disclaimer in the documentation
12   and/or other materials provided with the distribution.
13
14 THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
15 AND
16 ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
17 WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
18 DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE
19 FOR
20 ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
21 DAMAGES
```

18 (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;  
19 LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND  
20 ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT  
21 (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS  
22 SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

23

24 The views and conclusions contained in the software and documentation are  
those

25 of the authors and should not be interpreted as representing official  
policies ,

26 either expressed or implied , of the FreeBSD Project .